

2012 PM Challenge – EVA SE&I Submission

Title (max 8 words, include MBSE)

An MBSE Approach to Space Suit Development

Abstract (250 words)

The EVA/Space Suit Development Office (ESSD) Systems Engineering and Integration (SE&I) team has utilized MBSE in multiple programs. After developing operational and architectural models, the MBSE framework was expanded to link the requirements space to the system models through functional analysis and interfaces definitions. By documenting all the connections within the technical baseline, ESSD experienced significant efficiency improvements in analysis and identification of change impacts. One of the biggest challenges presented to the MBSE structure was a program transition and restructuring effort, which was completed successfully in 4 months culminating in the approval of a new EVA Technical Baseline. During this time three requirements sets spanning multiple DRMs were streamlined into one NASA-owned Systems Requirement Document (SRD) that successfully identified requirements relevant to the current hardware development effort while remaining extensible to support future hardware developments. A capability-based hierarchy was established to provide a more flexible framework for future space suit development that can support multiple programs with minimal rework of basic EVA/Space Suit requirements. This MBSE approach was most recently applied for generation of an EMU Demonstrator technical baseline being developed for an ISS DTO. The relatively quick turnaround of operational concepts, architecture definition, and requirements for this new suit development has allowed us to test and evolve the MBSE process and framework in an extremely different setting while still offering extensibility and traceability throughout ESSD projects. The ESSD MBSE framework continues to be evolved in order to support integration of all products associated with the SE&I engine.

Synopsis (50 words)

The ESSD SE&I team has utilized MBSE in multiple programs. By documenting all connections within the technical baseline, ESSD experienced significant efficiency improvements in analysis and identification of change impacts. ESSD MBSE framework continues to be evolved for supporting integration of products associated with the SE&I engine.

Biography (250 words)

Miriam Sargusingh is a NASA senior systems engineer with the Johnson Space Center (JSC) Extravehicular Activities (EVA) Office. She was the Architecture and Analysis Lead for the Constellation Program EVA Systems Project and currently works Systems Engineering and Integration (SE&I) for the EVA Systems Development team. She received a BS in chemical engineering from Yale University and a MS in Mechanical Engineering from National Technological University. Miriam has over 15 years experience in SE&I; more than 10 years in

2012 PM Challenge – EVA SE&I Submission

space suit development and sustaining engineering. She has accumulated more than 8 years of experience in model based systems engineering (MBSE) supporting various aerospace and aeronautical systems development projects. Miriam currently serves on the JSC Systems Engineering Forum planning team and the JSC MBSE splinter team.

Christine Kovich is a systems engineer from Wyle currently working with the Extravehicular Activity (EVA) Project Office. She has over 16 years of experience in the EVA-related discipline. Christine has 6 years experience in SE&I serving as the book manager for the EVA Systems Requirements Document (SRD). During this timeframe she has also had the unique SE&I experience of working both at the Constellation Program level as well as in the EVA Systems Project Office. In addition she supports EVA's MBSE process development. Christine received a BSME from Tri-State University and her previous experience includes the following: project management, flight analog studies lead, ISS and Shuttle EVA flight hardware certification, and ISS EVA development hardware testing coordination/implementation.

Lauren Cordova is a systems engineer from Booz Allen Hamilton with over 2 years experience as the Book Manager for the EVA Ops Con, Architecture Description Document, and Functional Analysis Document. In addition she supports EVA's Model Based Systems Engineering (MBSE) process development and implementation as well as advanced EVA systems development. Lauren received a mechanical engineering degree from the University of Texas at Austin, and her experience includes SE&I, manufacturing, quality engineering, and environmental compliance across the aerospace, medical, and utility industries.



An MBSE Approach to Space Suit Development

L. Cordova, C. Kovich, M. Sargusingh



Agenda



- Introduction
- Model Based Requirements Development & Management Approach
 - Ops Con/models
 - Environments
 - Architecture
 - Interfaces
 - Functions
 - Decomposition
 - Performance Measures
 - Requirements / Verification
 - Functional Analysis
 - Technical Planning
 - Documentation
- Conclusion

Introduction



- What is Model-Based Systems Engineering?
 - Model-Based Systems Engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases (INCOSE-TP-2004-004-02, Version 2.03, September 2007)
- Why is it important?
 - Ensures traceability and completeness of systems engineering products (e.g., Ops Con, Requirements, Verifications, etc.)
 - Is an approach to systems engineering where information about the system is defined, standardized, and interdependent during the lifecycle process
 - Is a key step in moving from a “document-based” configuration management approach to a “data object or model-centric” approach

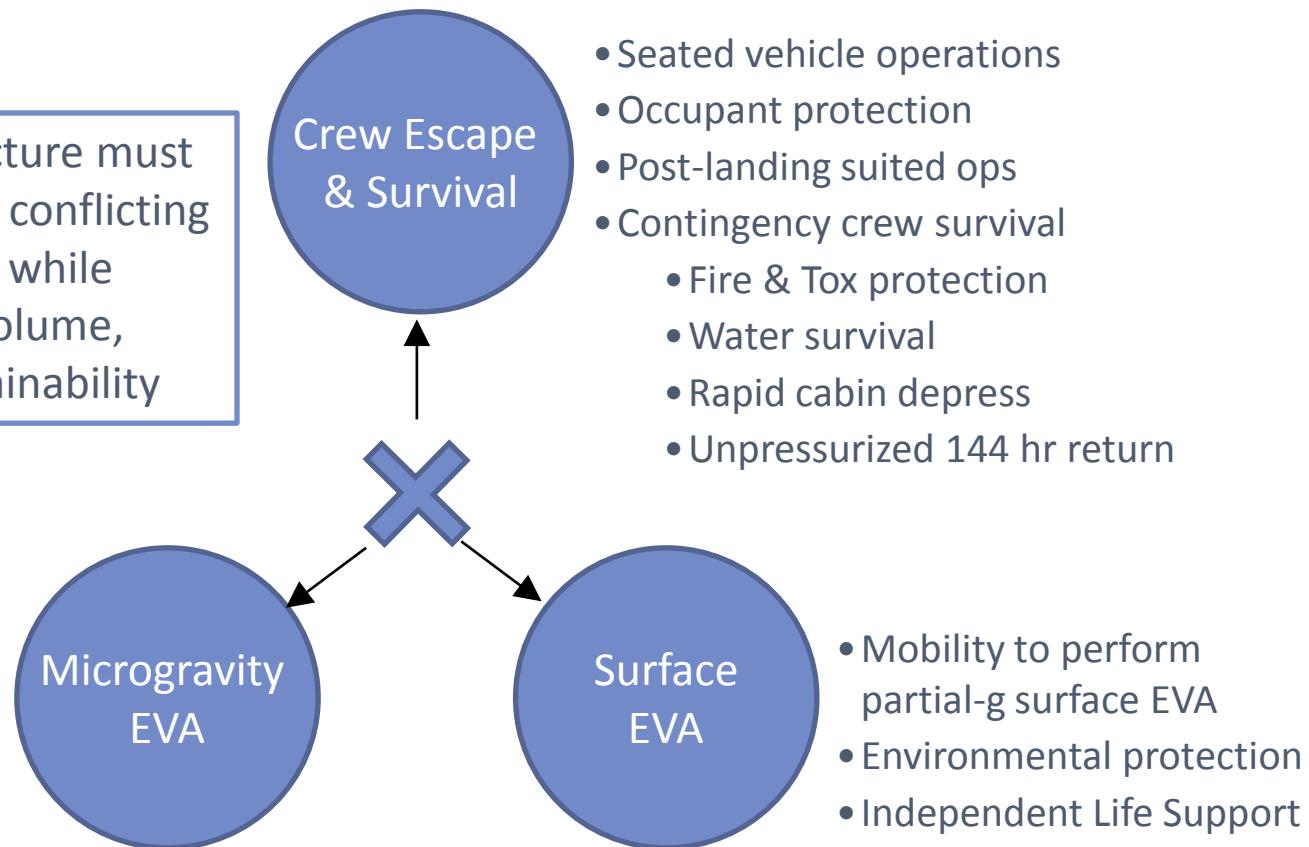
Overview



- The NASA Johnson Space Center EVA Office Systems Engineering and Integration (SE&I) team utilizes MBSE to develop and manage requirements for an evolving, complex system that spans multiple programs

The space suit architecture must meet various and often conflicting mission objectives while optimizing mass, volume, reliability and maintainability

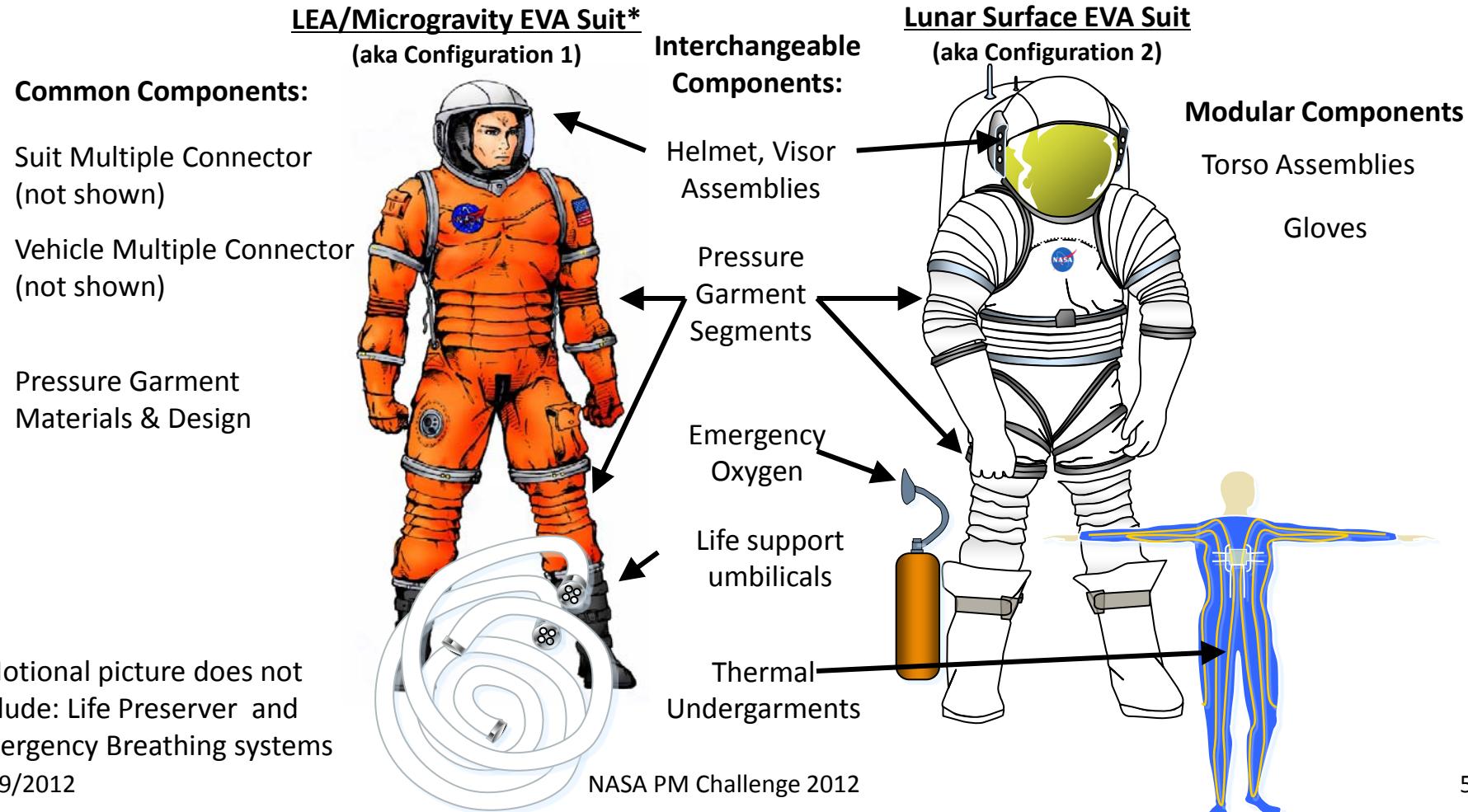
- Pressurized μ -g mobility
- Environmental protection
- Vehicle supported or independent life support
- Vehicle Maintenance/Reconfiguration



Space Suit System (EVA System Reference 2)



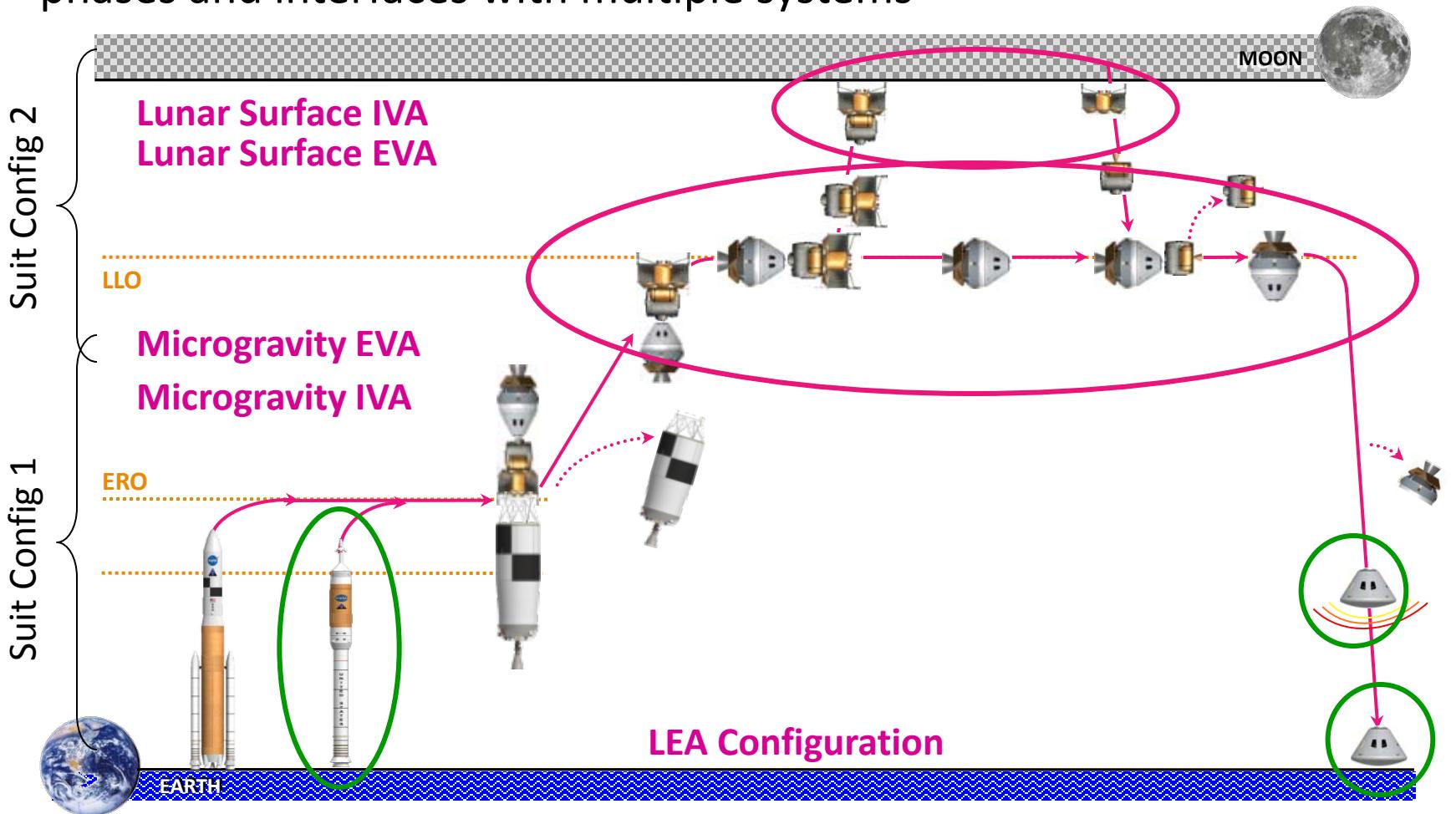
- The original design solution developed for Constellation involved a modular, reconfigurable, component-based architecture



Lunar Crewed Mission



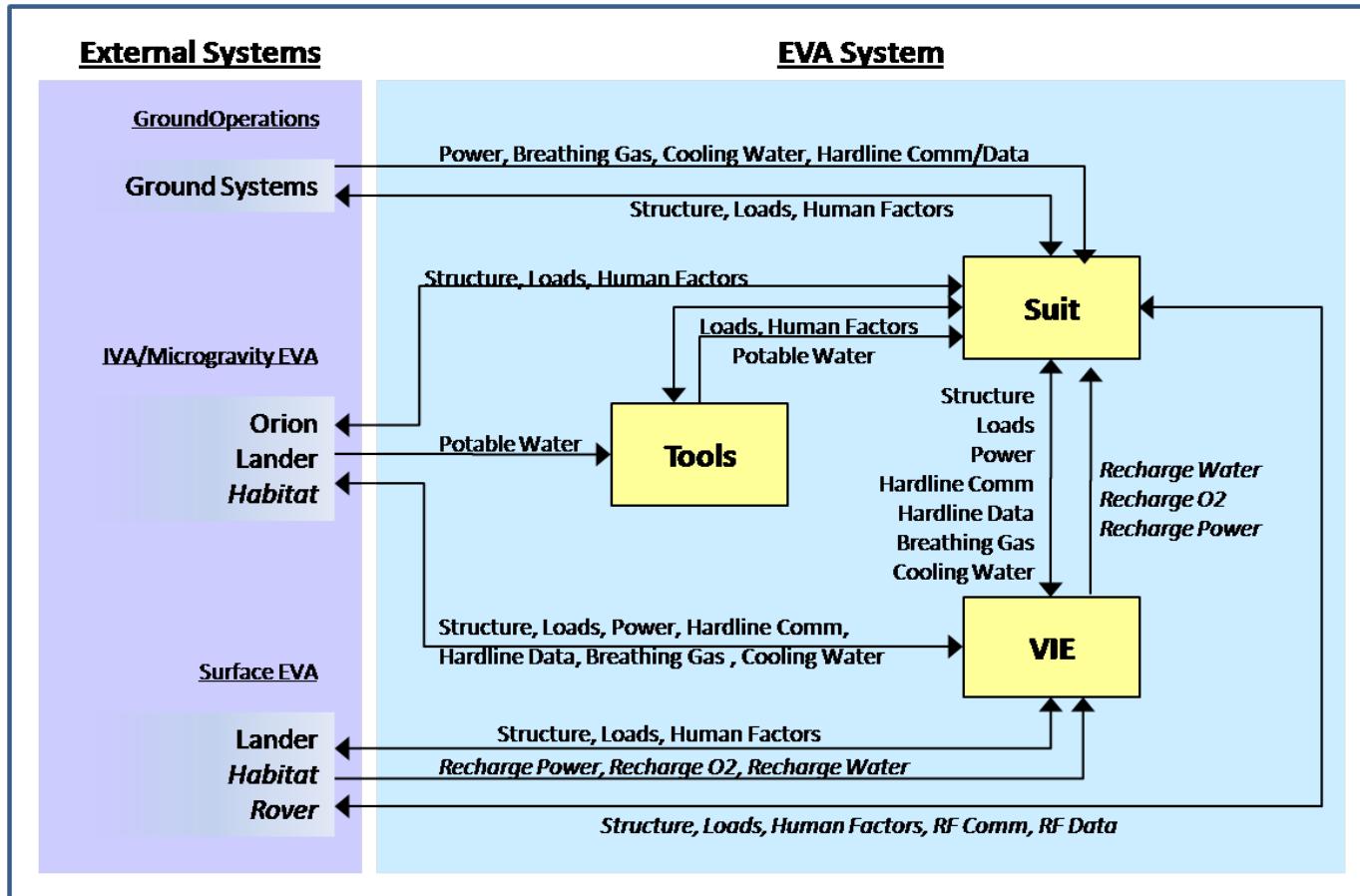
- In the Constellation Program Lunar Design Reference Mission (DRM), the suited crewmember, and therefore the space suit, is involved in all phases and interfaces with multiple systems



Space Suit Interfaces



- The space suit interfaced with every crewed vehicle included in the Constellation architecture





MODEL BASED REQUIREMENTS DEVELOPMENT & MANAGEMENT APPROACH

Implementation of the SE Engine

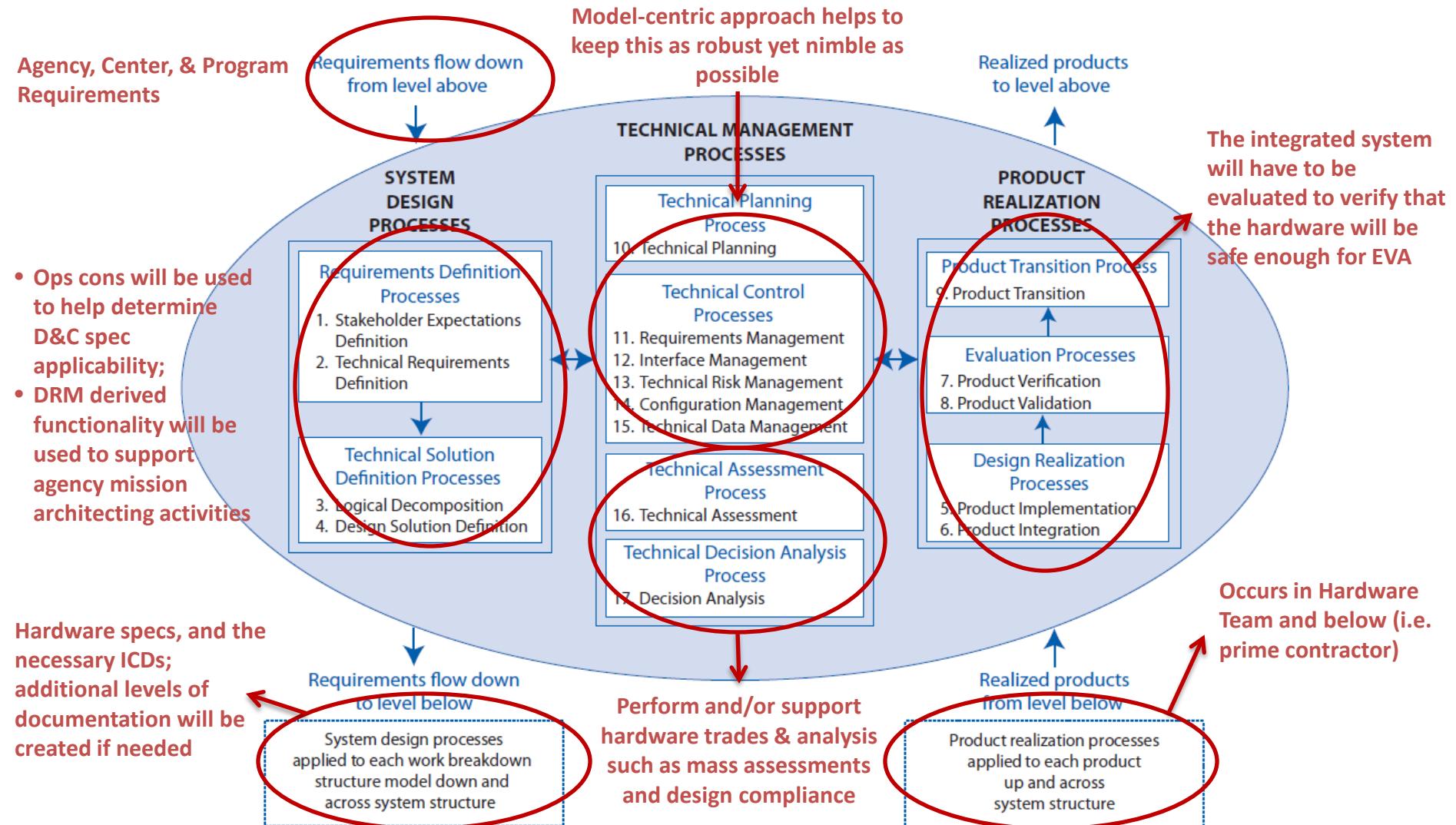
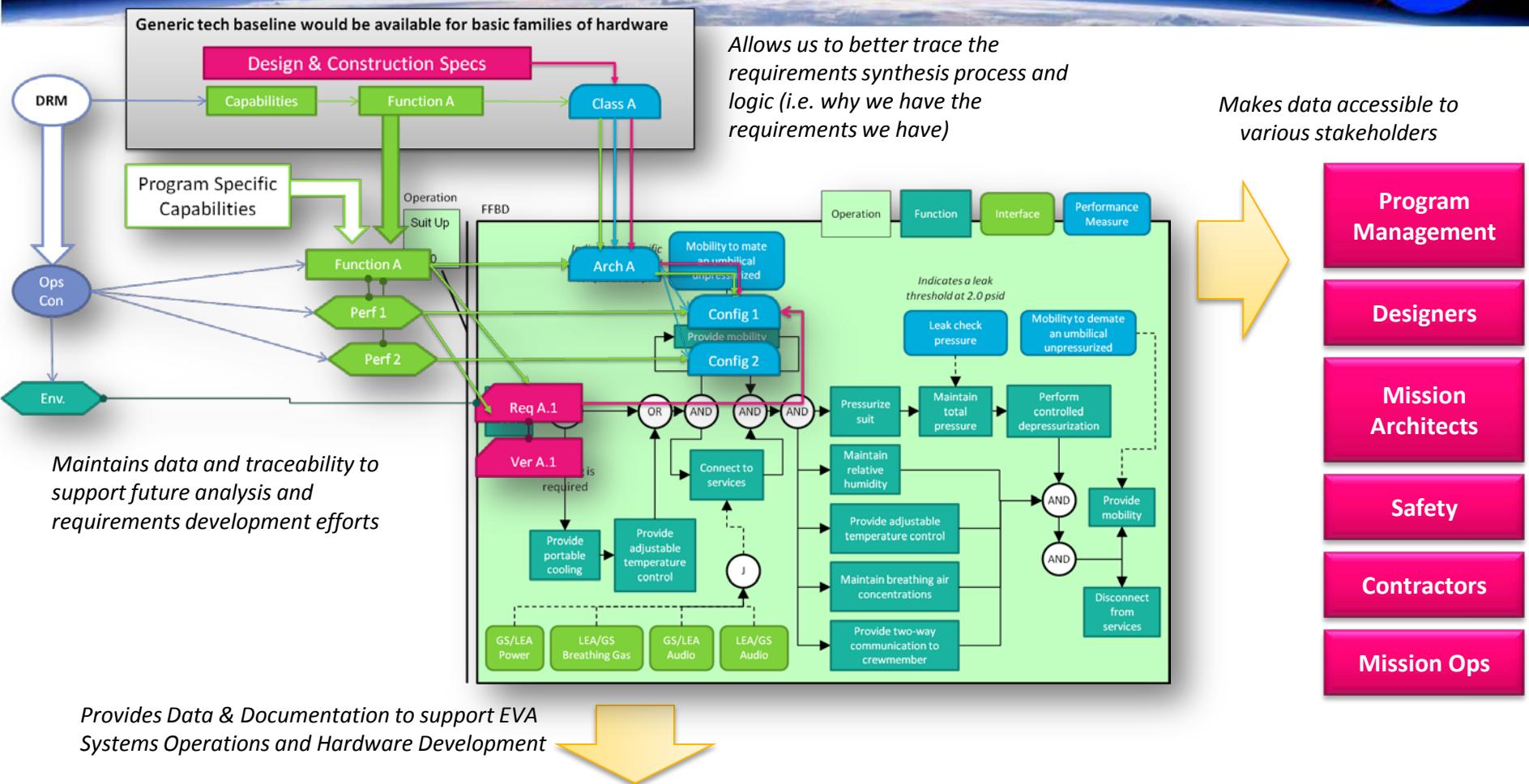


Figure 2.1-1 The systems engineering engine

Figure: NASA/SP-2007-6105 Rev1

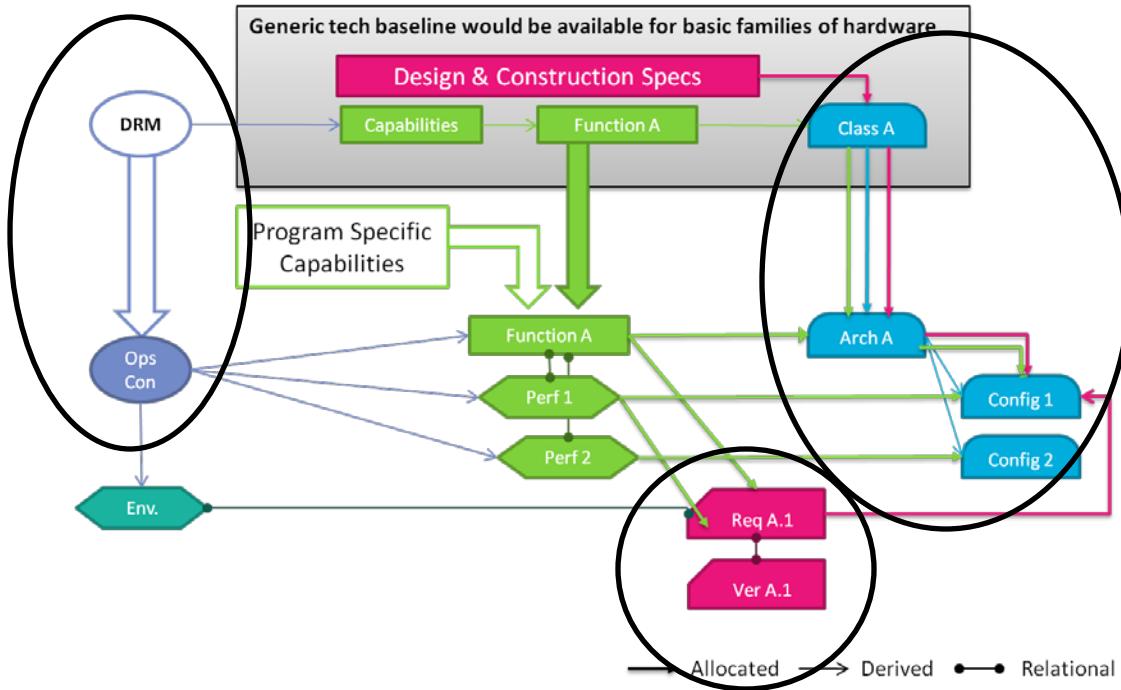
Model Based Data Infrastructure



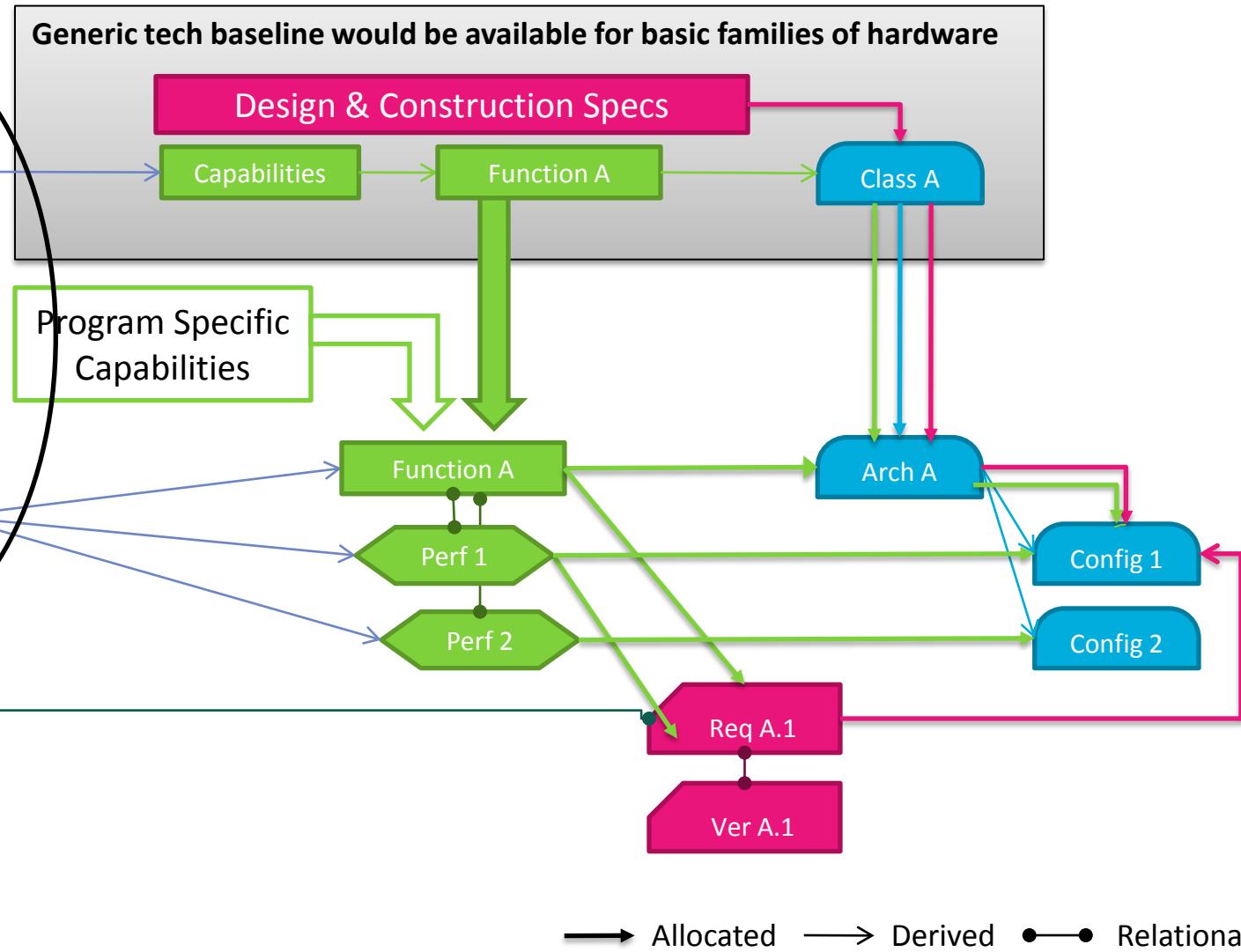
System Model Framework



- The Space Suit System Model includes definitions of the three basic parts of the technical baseline (aka the Tri-Force): operational concepts, architecture/design, and requirements
- By capturing all of the connections between the technical baseline, we experienced a significant efficiency improvement in analysis and identification of change impacts



Operational Concepts

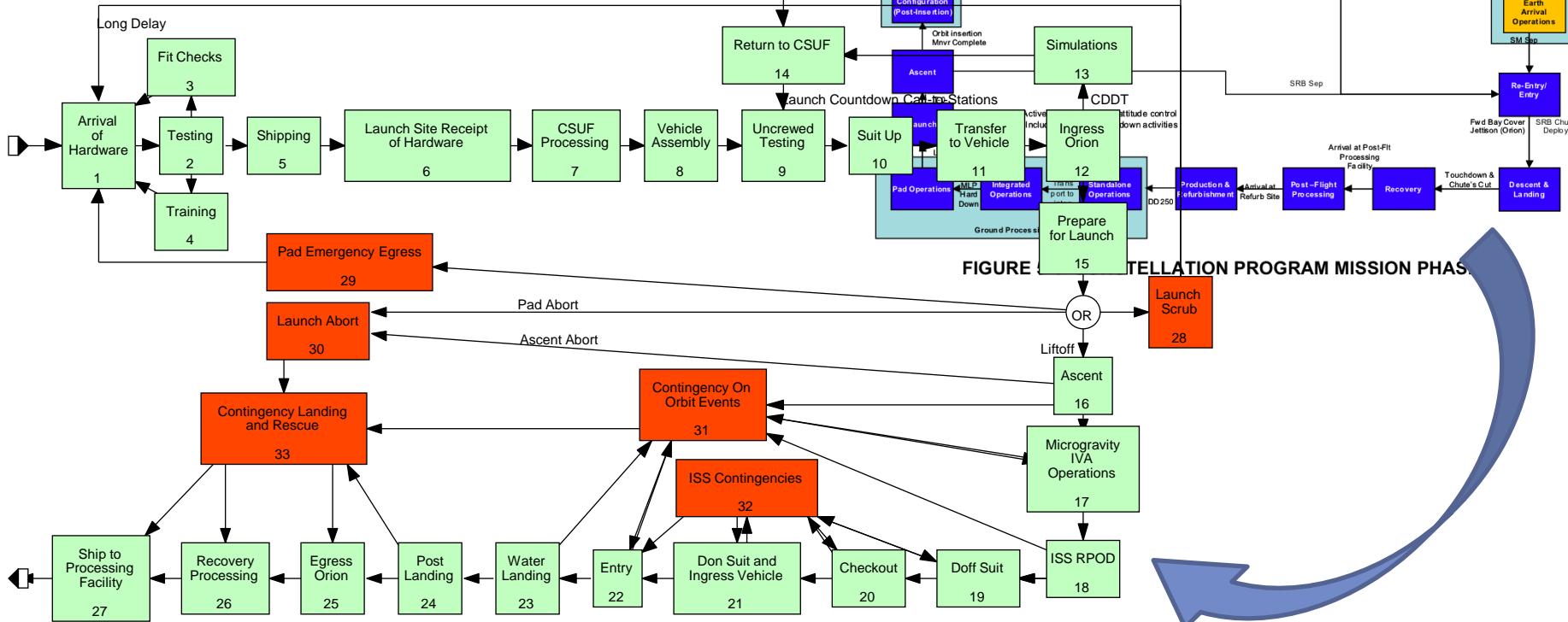


→ Allocated → Derived •• Relational

Operational Models



- The Space Suit DRM decomposes program-level operations into suit-focused activities and adds contingency operations/transitions

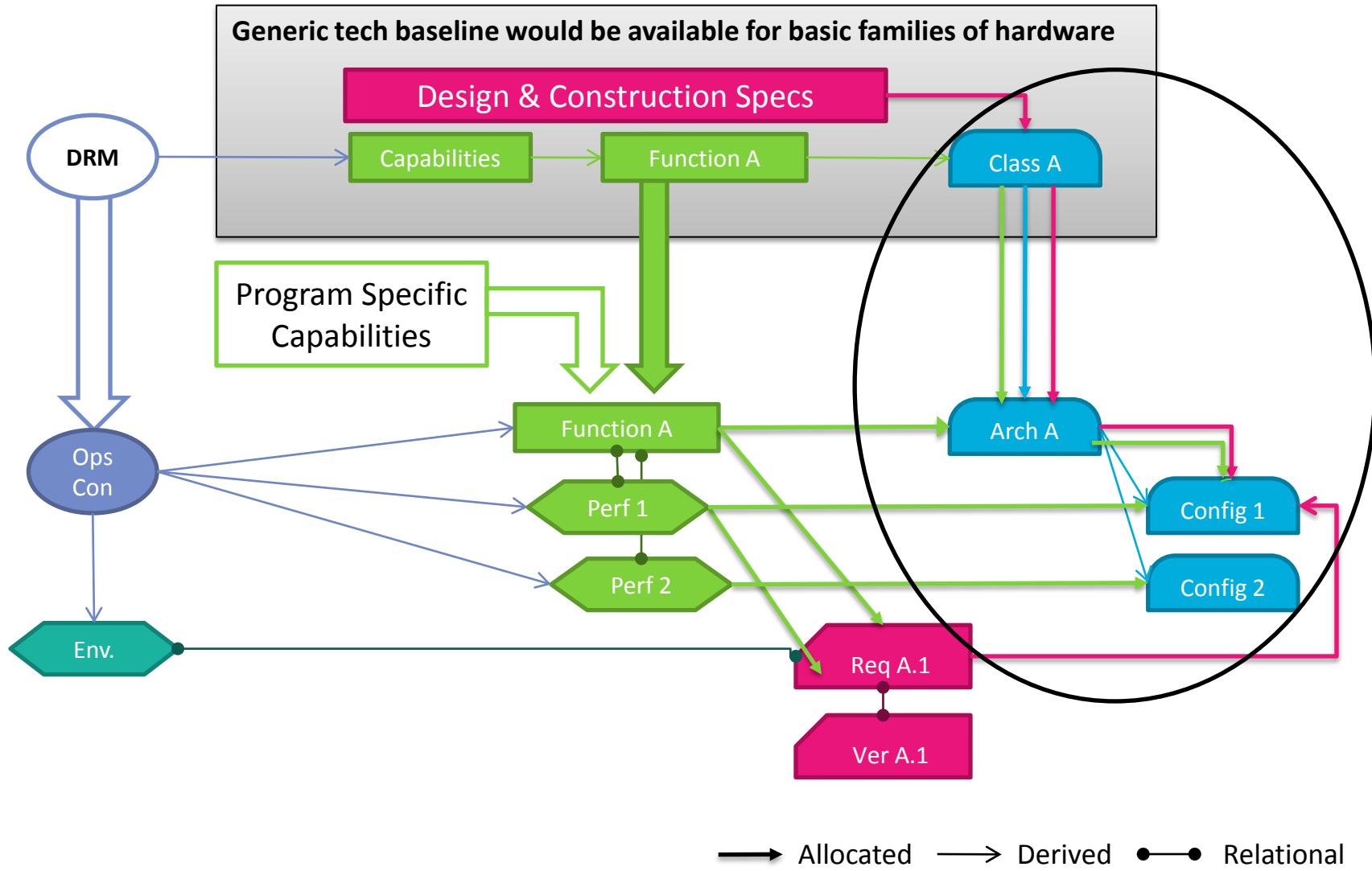


Operational Concepts



- The Space Suit Operational Concepts (Ops Con) serve as the basis for communication between our stakeholders and designers
 - Stakeholders help us to understand their expectations on what the suit capabilities are and how it will be used
 - Designers provide information on how the suit is expected work
- Operational concepts are currently presented in document format:
 - Multi-Purpose Crew Vehicle (MPCV) Flight Suit Element Operational Concept provides Ops Con associated with Block 0 and Block 1 MPCV operation
 - International Space Station (ISS) Detailed Test Objective (DTO) Design Reference Document contains Ops Con associated with the Extravehicular Mobility Unit (EMU) Demonstrator
 - Advanced EVA Operations will contain the Ops Con associated with the Capability Driven Framework DRMs

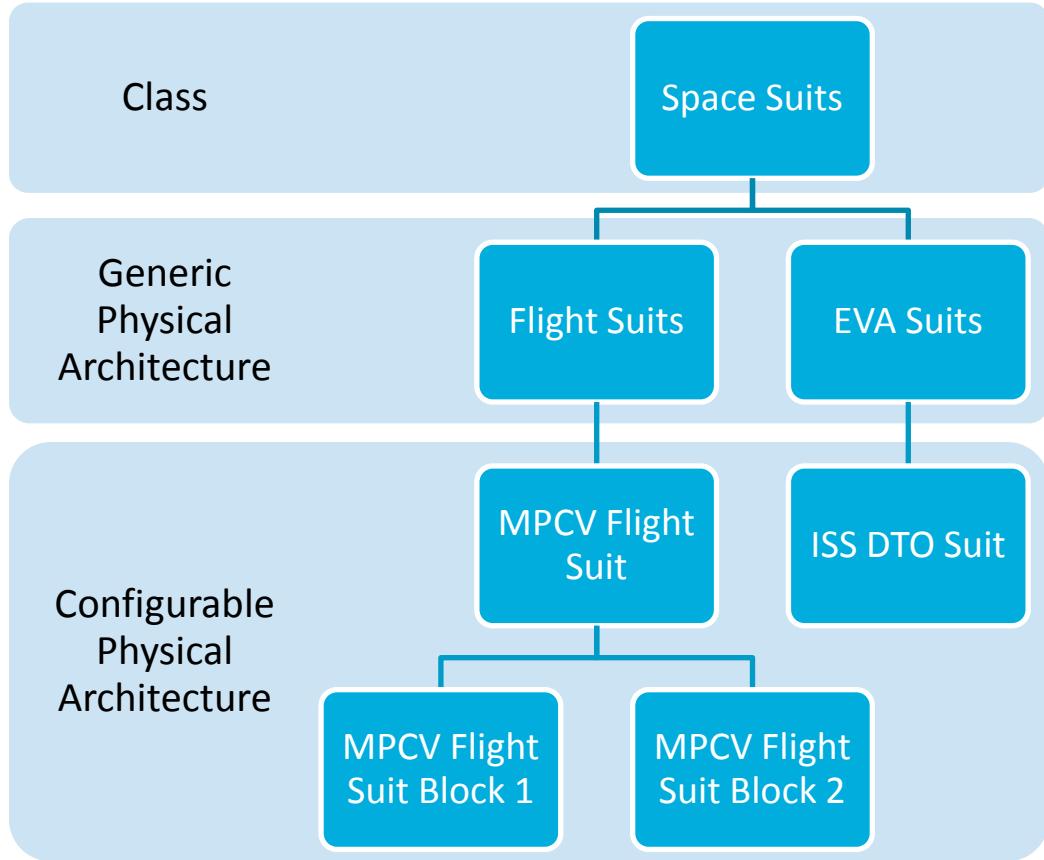
Architecture



Hardware “Classes”



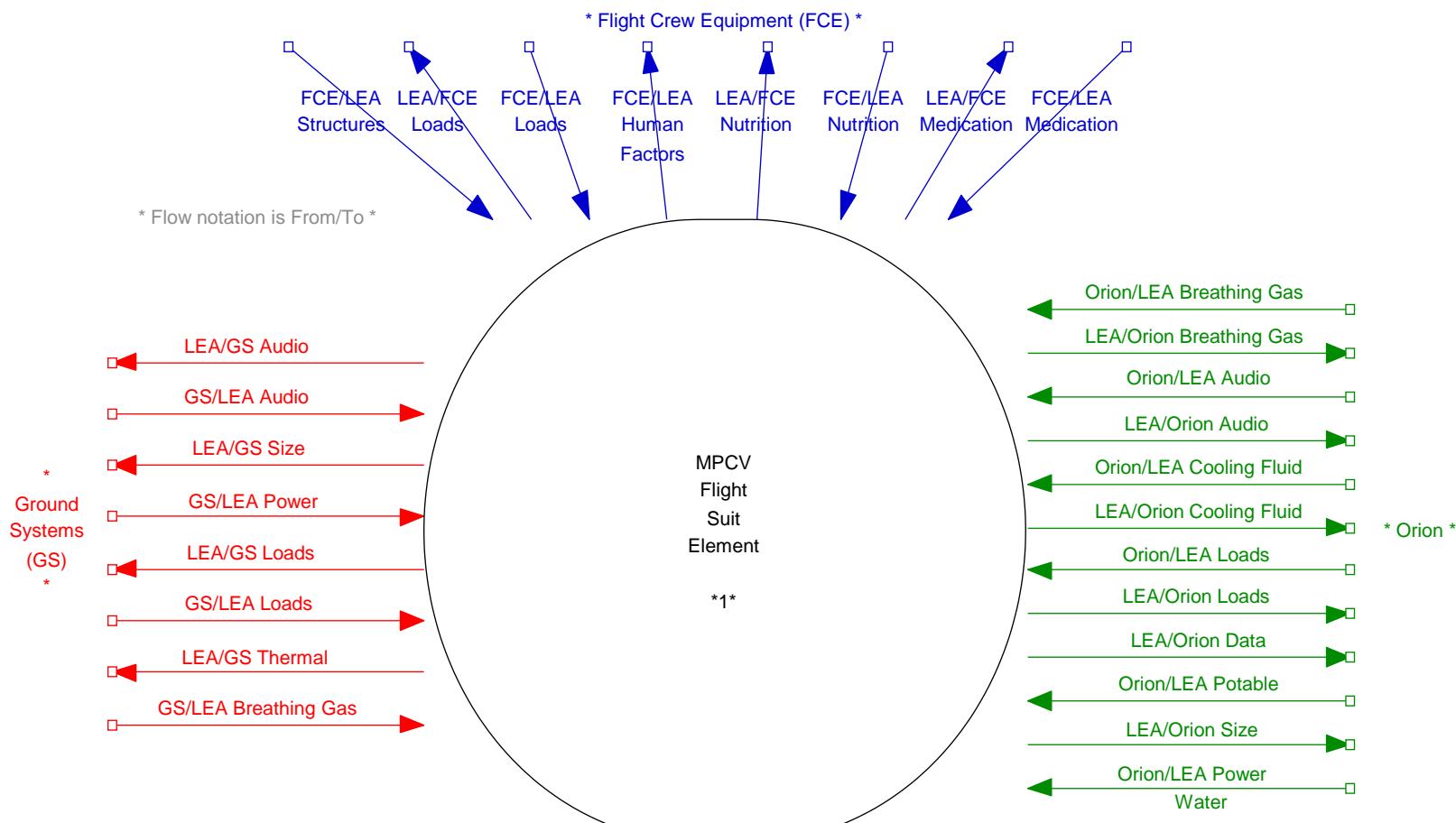
- The “class” hierarchy defines the scope of the project
- A “class” of hardware defines a generic technical baseline
 - Default set of standards & specifications
 - Supports a generic set of operational scenarios
 - Generic functions
 - Generic hardware
- Provides a platform for mission specific systems engineering



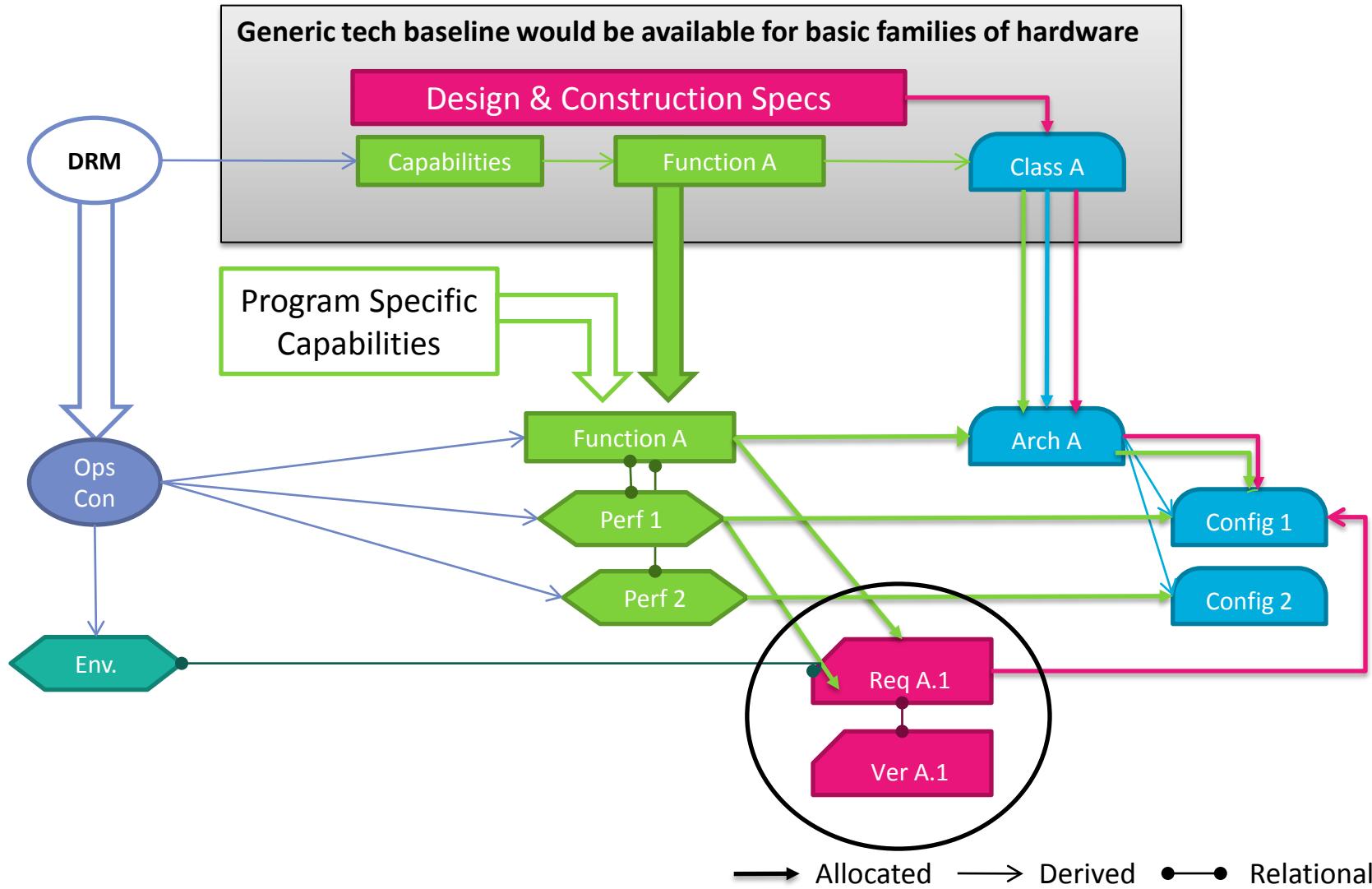
Interface Management



- The Architecture is further defined with Physical Architecture Diagrams (PADs) showing the system interfaces



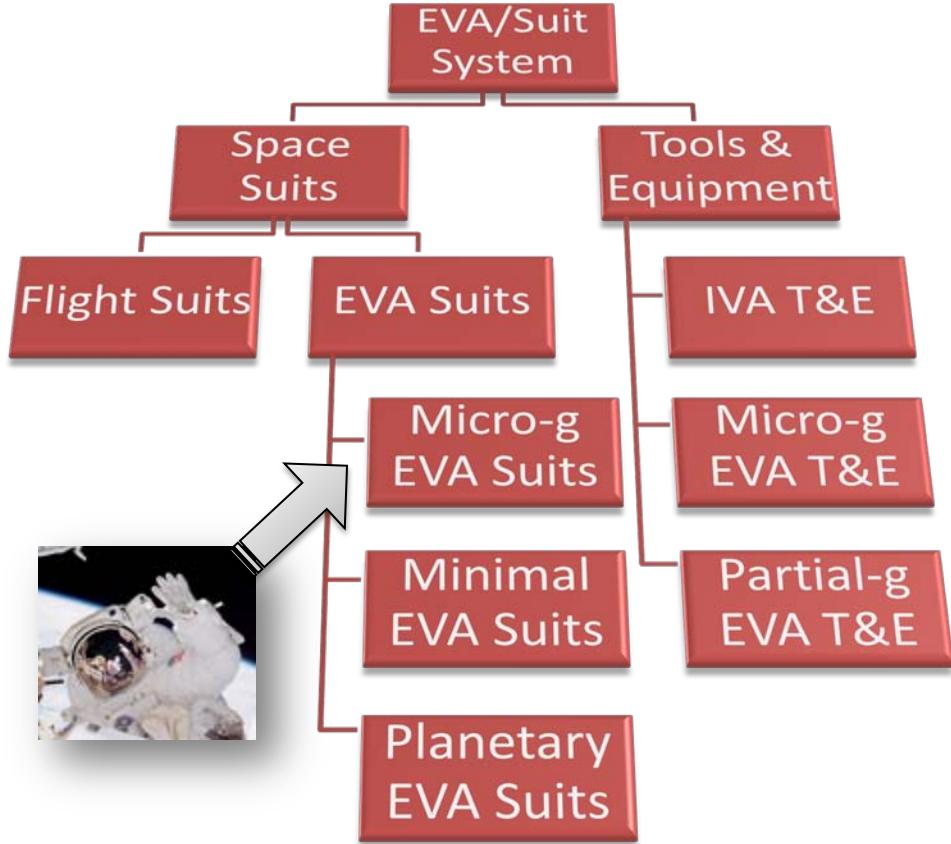
Requirements



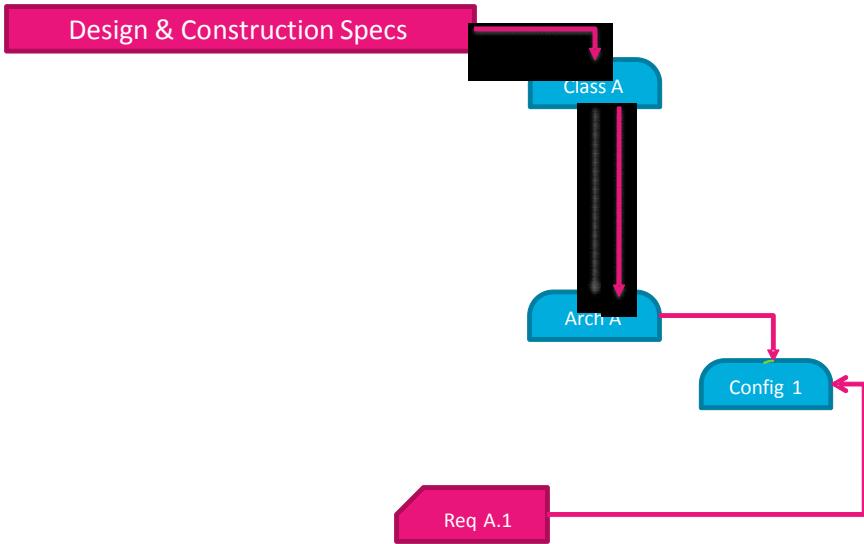
Requirements – SRD Structure



- The class hierarchy produces a requirements set that is easily extensible to future suit developments
 - Establishes a set of generic space suit reqs/standards
 - Builds upon generic reqs by allowing for vehicle/mission specific reqs to be captured
- Linking requirements to higher level classes limits “starting from scratch” when new programs are stood up
- This structure was created to maintain horizontal integration across all suited efforts



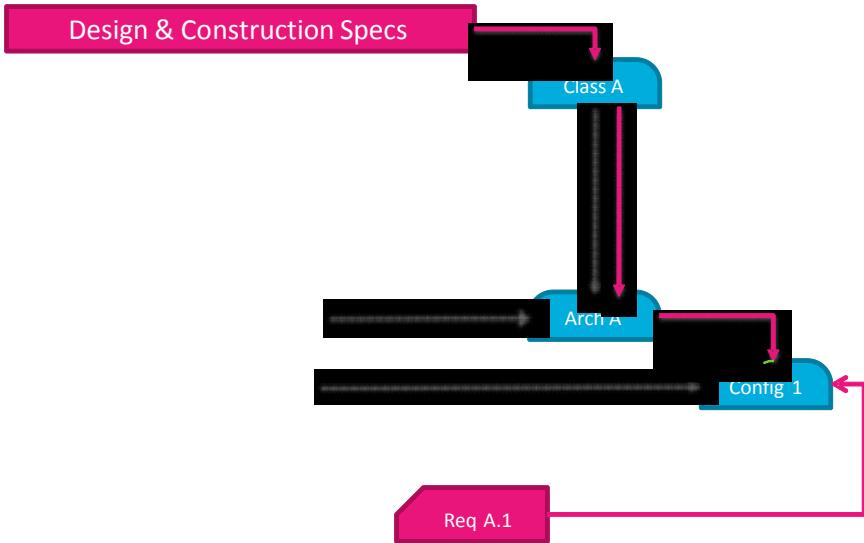
Derived Requirements



Standard:

The WMS shall prevent direct contact of contained urine with the crew's skin

Derived Requirements



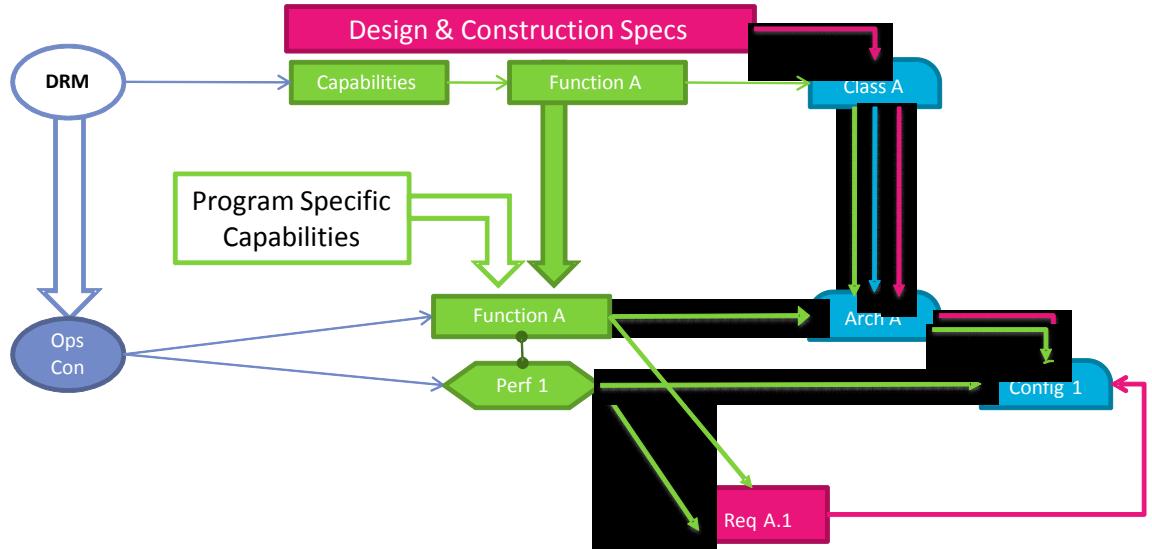
Standard:

The WMS shall prevent direct contact of contained urine with the crew's skin

Physical constraint:

The 12-hr MAG shall fit within the allocated stowage volume.

Derived Requirements



Standard:

The WMS shall prevent direct contact of contained urine with the crew's skin

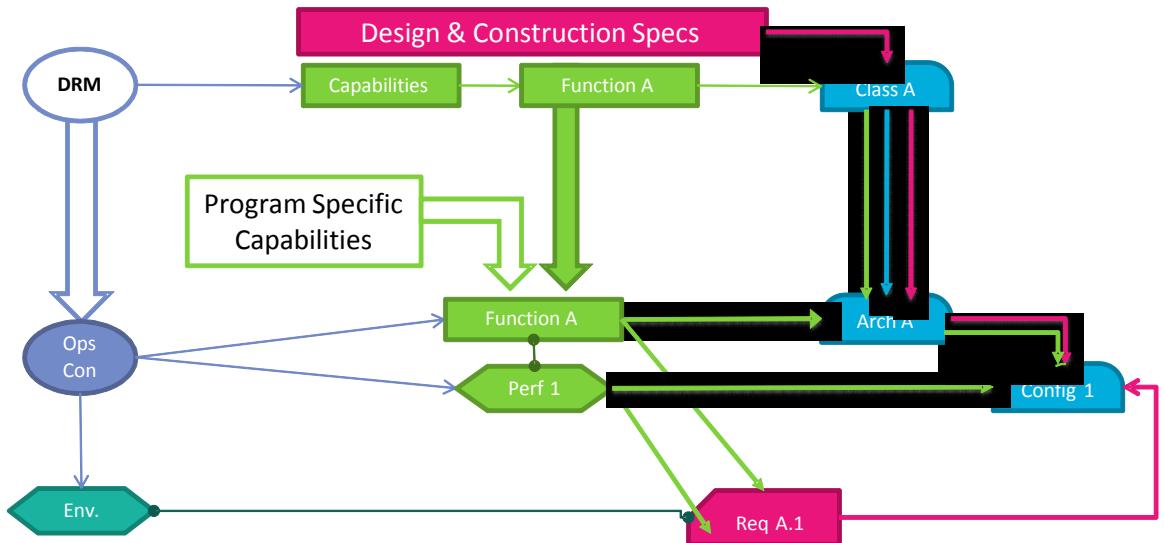
Physical constraint:

The 12-hr MAG shall fit within the allocated stowage volume.

Functional/Performance:

The MPCV Flight Suit shall contain urine for 12 hours during Flight Test Missions

Derived Requirements



Standard:

The WMS shall prevent direct contact of contained urine with the crew's skin

Physical constraint:

The 12-hr MAG shall fit within the allocated stowage volume.

Functional/Performance:

The MPCV Flight Suit shall contain urine for 12 hours during Flight Test Missions

Environment Definition:

The WMS shall function after stowage in the Orion pressurized volume

Verification



- Verification planning
 - Begins early in the project life cycle
 - Updates to verification planning will continue throughout the logical decomposition and design development phases, especially as design reviews and simulations shed light on items under consideration in the requirements development phase
- Our verifications were drafted along with the requirements to ensure that we had verifiable requirements
 - All verifications are linked to their respective requirement
 - The Environments map to specific operations from the Ops Con model which was intended to support the verification planning
 - Certification is a classification applied to environments if the verification for the requirement calls for 'certification' in a specified environment

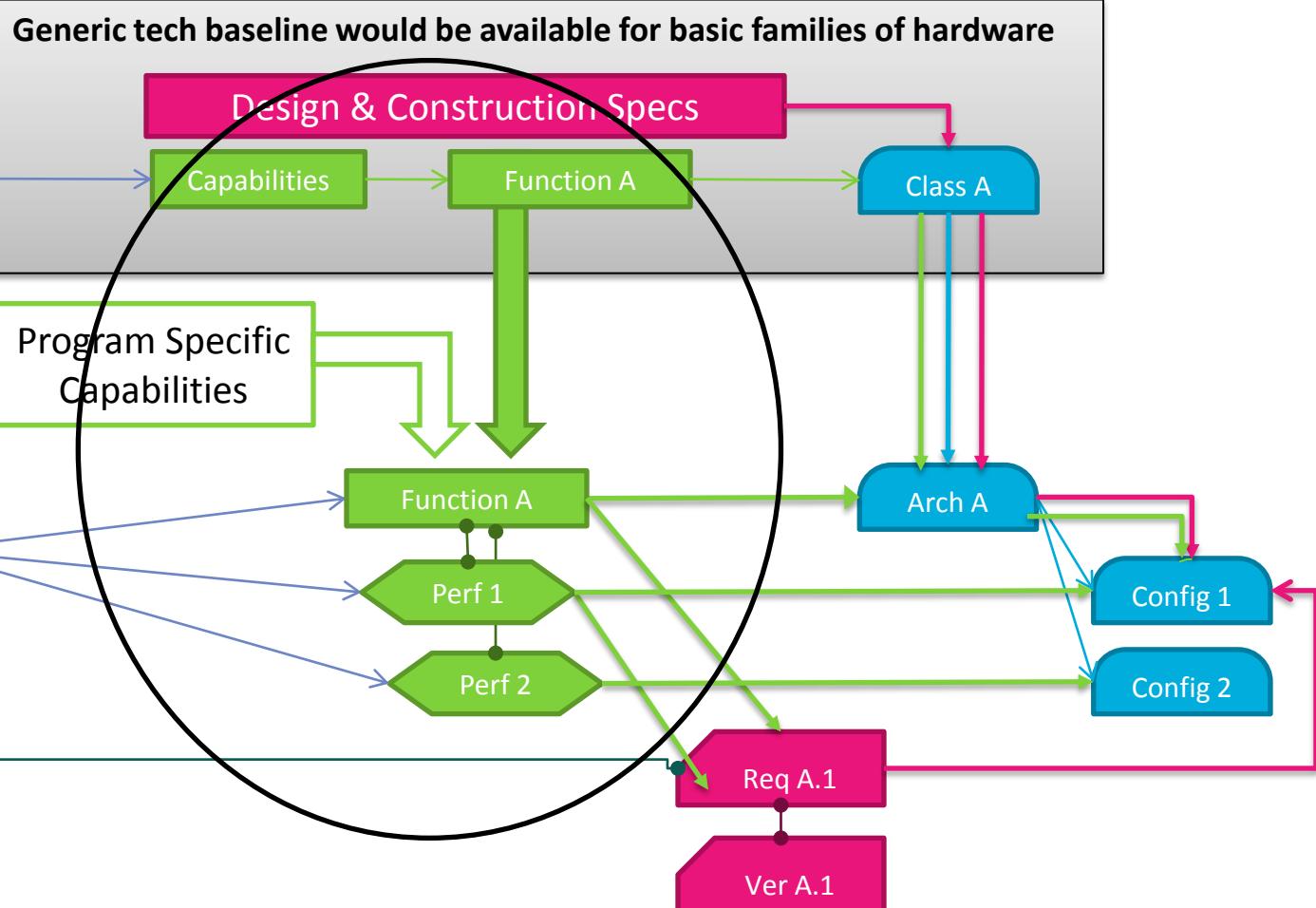
Without a verifiable baseline and appropriate configuration controls, later modifications could be costly or cause major performance problems for your system

Verification Roles & Responsibilities



- A need to clarify verification roles and responsibilities was realized during discussion regarding requirements that were applicable to the prime contract:
 - Which party is responsible for developing detailed verification objectives (DVO)
 - Which party is responsible for executing per the DVO
- Approach
 - Responsibility categories were developed to group verifications of similar scope and complexity. Establishing categories (buckets) ensured that approaches were consistently applied and that any exceptions to the philosophy were noted
- Assumptions
 - Requirements/children sharing equivalent scope will not duplicate verification efforts (this is based on similar/equal requirements residing at multiple levels)
 - Requirements sharing equivalent scope and functionality (noun changes) need to share a verification approach

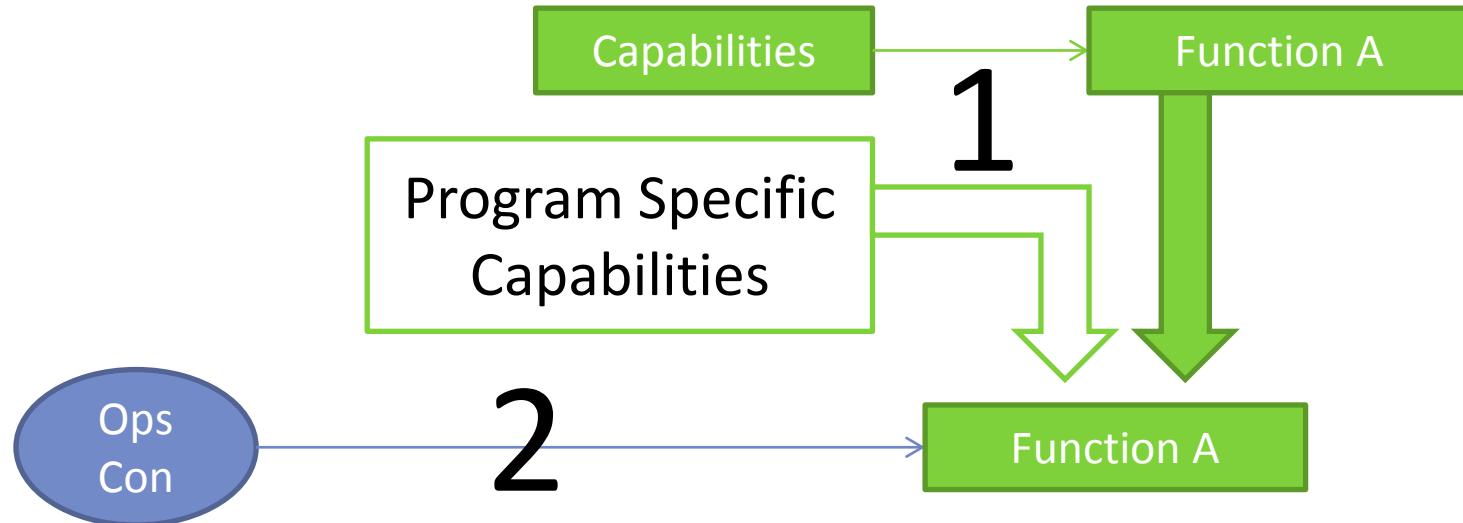
Functions



Functional Decomposition



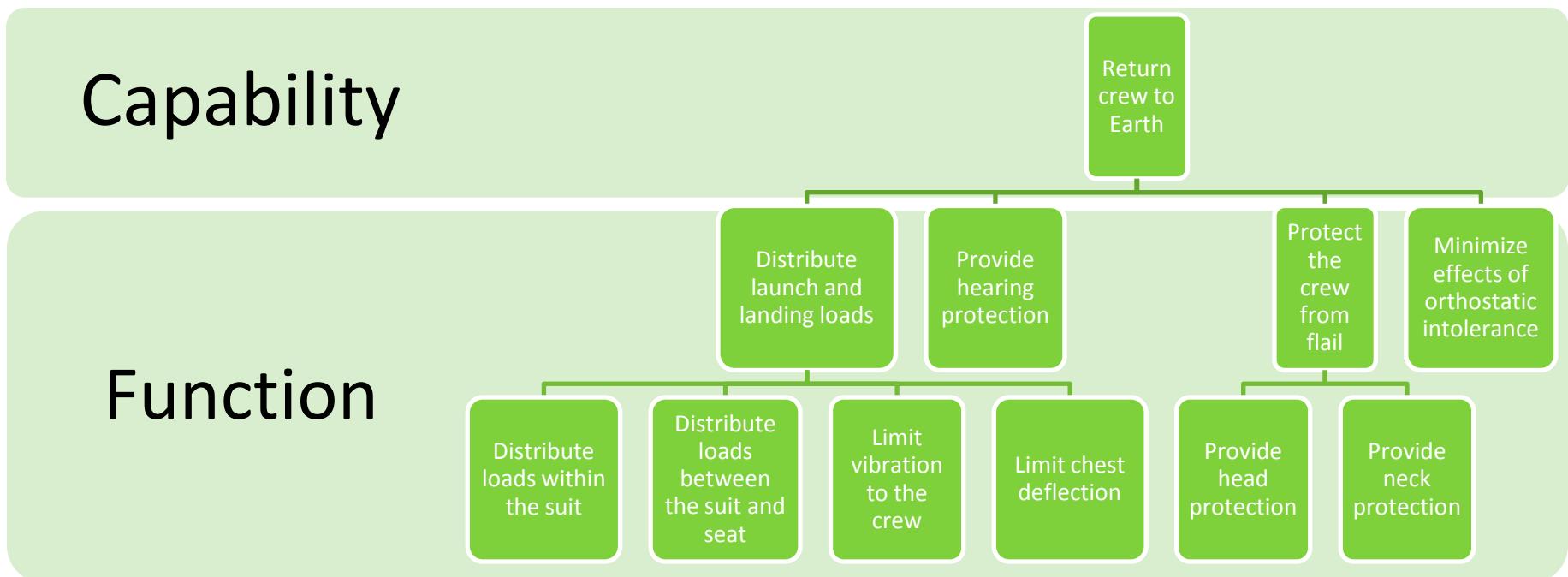
- Functions provide the link between operations, architecture, and requirements (Tri-Force)
- Functions were created using two methods:
 1. Decomposition from Program allocated “capabilities” through a hierarchical approach
 2. Derivation from the operations



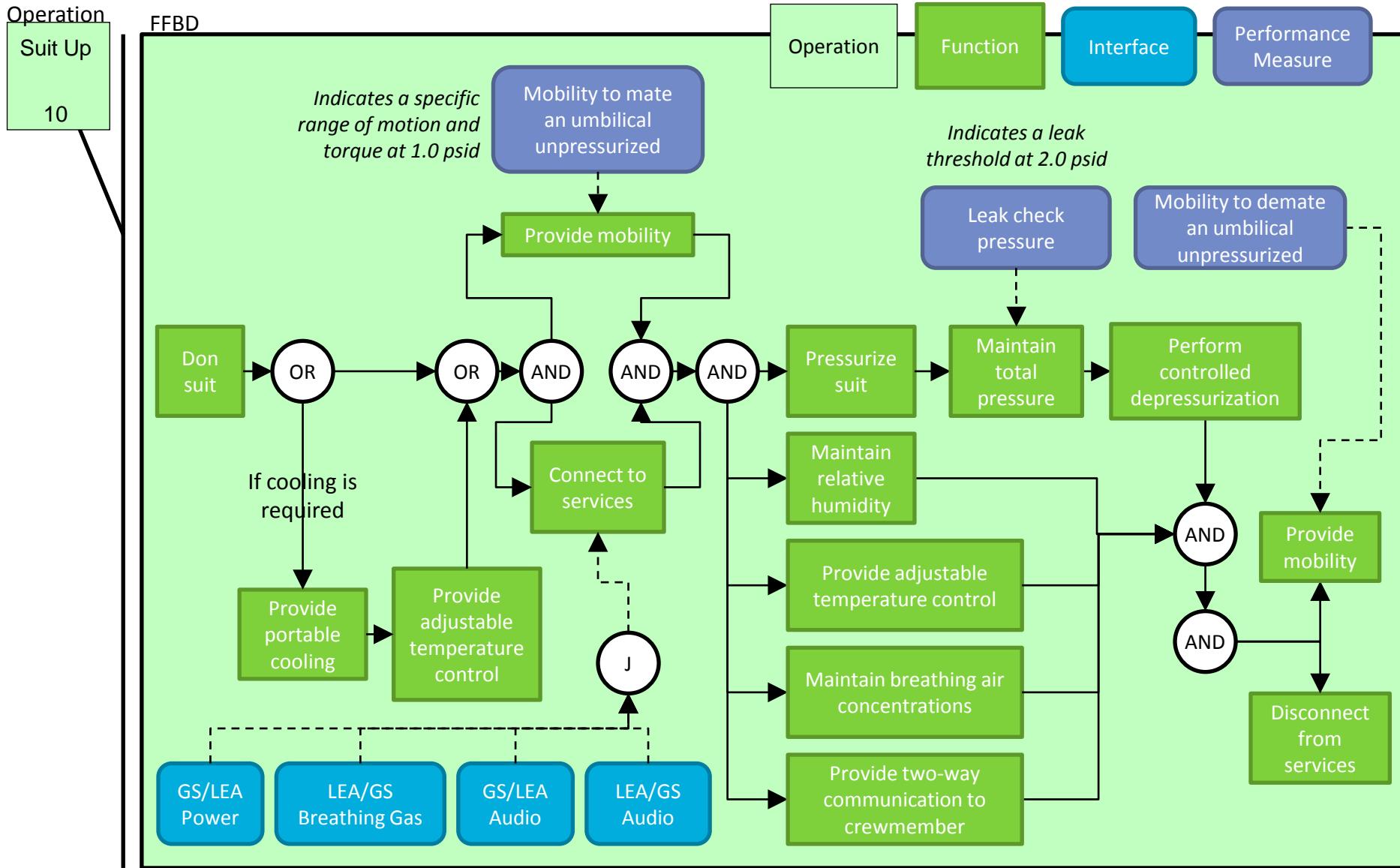
Decomposition from Capabilities



- Capabilities are high level/programmatic functions
 - Capabilities decompose into lower level functions specific to the system of interest
 - Capabilities were included to show traceability to Program allocated functionality



Derivation from Operations

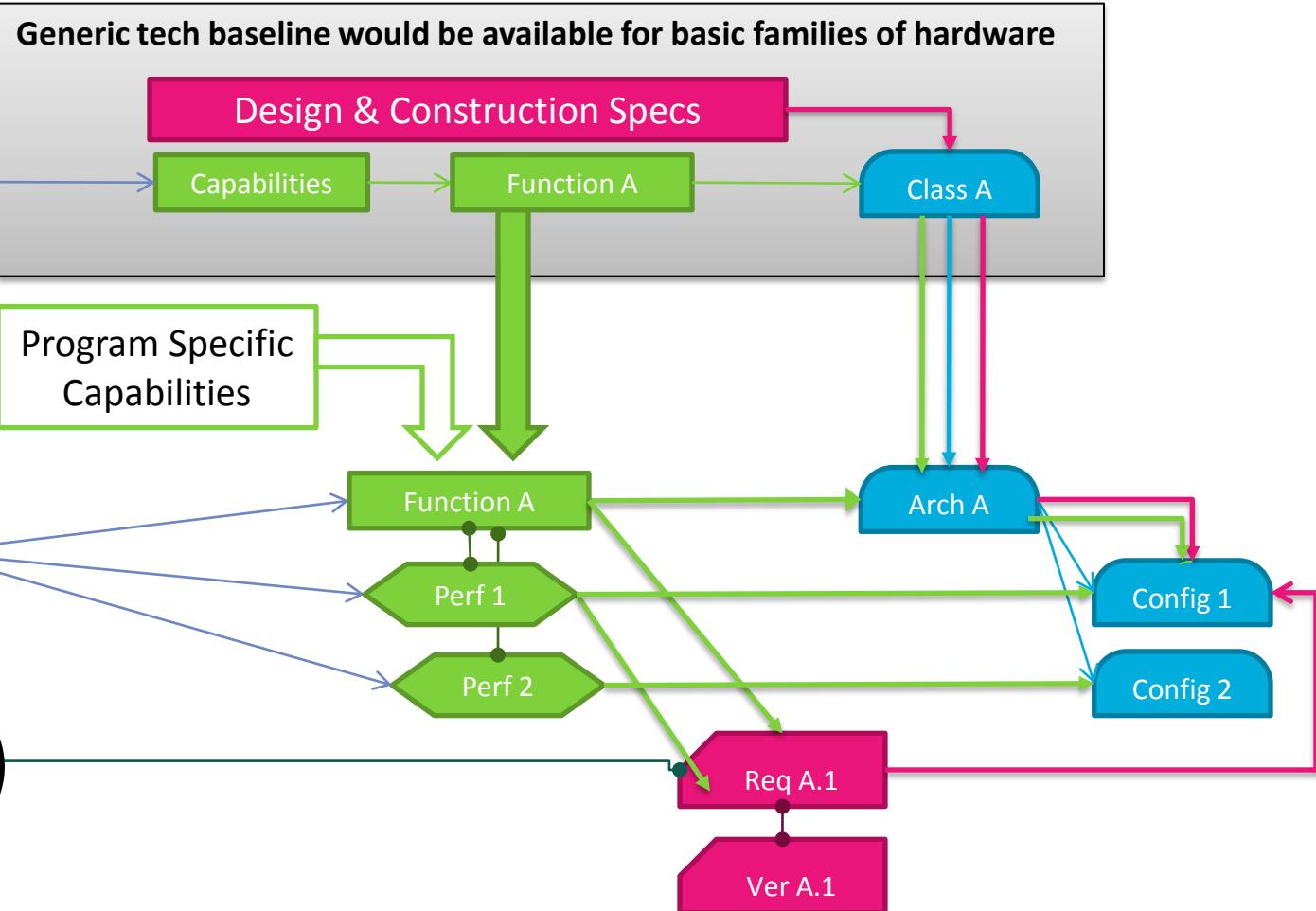


Requirements Validation



- Requirements were initially developed from Subject Matter Expert (SME) input and decomposition/flow down of program allocated requirements
 - Though the Ops Con was used to derive some requirements, they were not initially linked
- Audits were used to show that program allocated requirements were answered in our requirements set (traceability report)
- The “Tri-Force” allowed for validation of the requirements set from a functional perspective (did we have the right requirements)
- After developing the MBSE framework, the requirements were linked to the System level models through the functional analysis and interface definitions
 - This aided in identification of impacts for a proposed change to the technical baseline

Environments



→ Allocated → Derived •—• Relational

Environments



- The system model framework included several aspects of the space suit environments:
 - Definition of environments to which the hardware could be subjected
 - Definition of environments in which the hardware needs to operate
 - Definition of environments to which the hardware would be optimized
 - Requirements specifying the environments to be created by the space suit
- The space suit environments definitions were scattered across several documents with no mapping to operational concepts, requirements or capabilities
 - We were able to publish the definitions in 1 document
- Environments map to specific operations from the Ops Con model
 - Using the definition to Ops Con mapping, the environments could be traced to requirements and verifications

Technical Planning



- We also used Issues and Actions to aid in technical planning
 - Issues were comprised of those items considered To Be Determined (TBD) and To Be Resolved (TBR)
 - Actions were items used to track open work that was not part of the TBD/TBRs but were necessary to update the technical baseline
 - Both Issues and Actions were associated with the following items:
 - Ops Con
 - Architecture
 - Functions
 - Requirements
 - Verifications
 - Environments

Documentation



- Templates were created to generate documents with particular formatting
 - Templates were hardcoded to pull in particular document sections with associated linked data
- Documents were generated by printing out specified items based on the associated links (e.g. requirements linked to architecture items)
 - Without changing our current structure, the linkages also allowed us to create documents for specific DRMs or suit configurations (e.g. SRD, ERD, contract applicable requirements)
- Data can also be extracted using a query
 - Exports can be performed in csv, rtf, or HTML formats

Other models



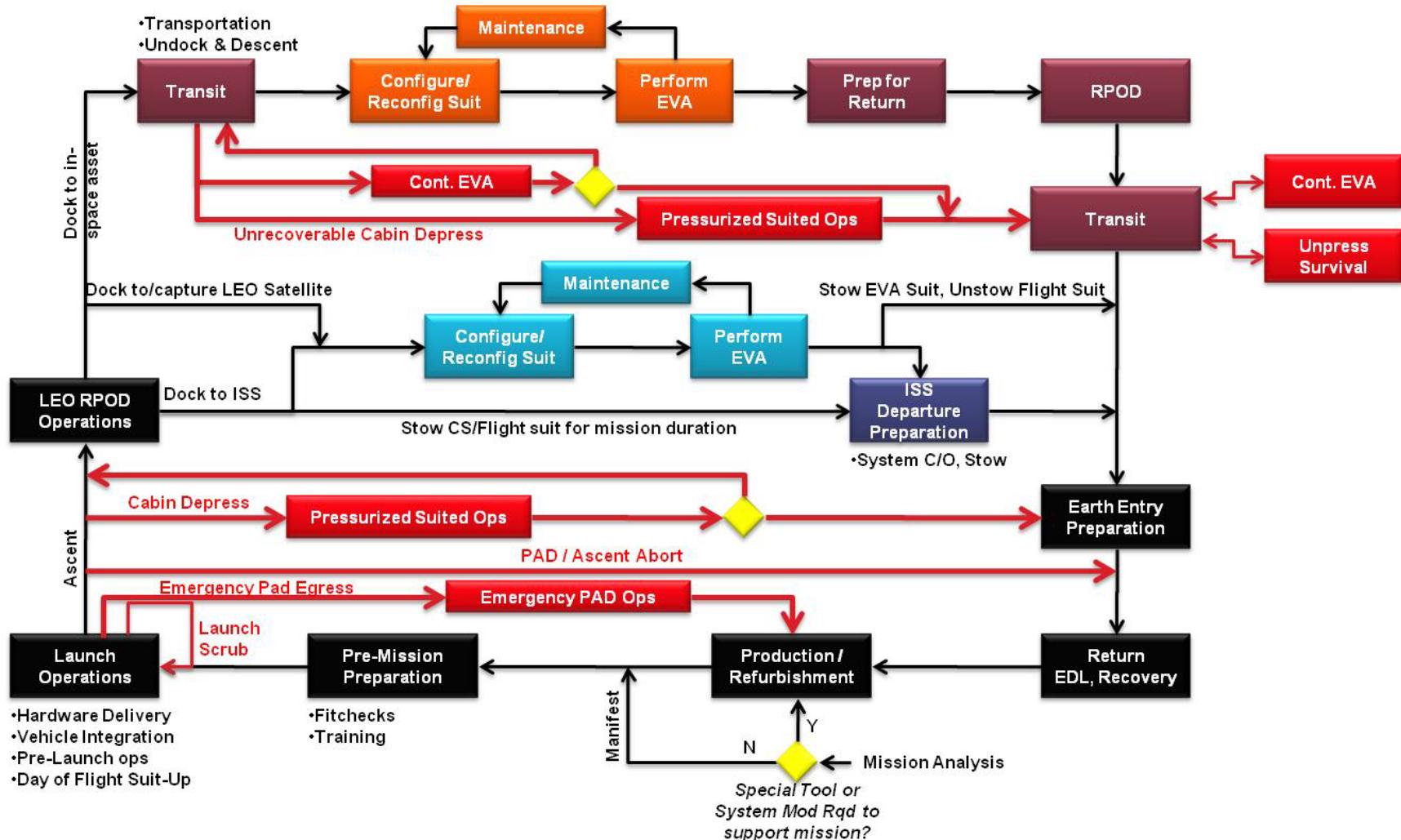
- Several models have been used to develop and validate system requirements
 - Wissler Model
 - Thermal Desktop CFD Model
 - Excel Pressure Drop / Flow Model
 - Macroflow Pressure Drop Model
 - EVA System Mass Tracker
- These models are configuration managed (i.e. data managed) by the analyst (except Wissler)
 - Model descriptions and results are presented to peers and stakeholders as part of analysis process
 - Results are evaluated and used to support requirements validation or change package
- There are no direct links between analytical models and SE Database



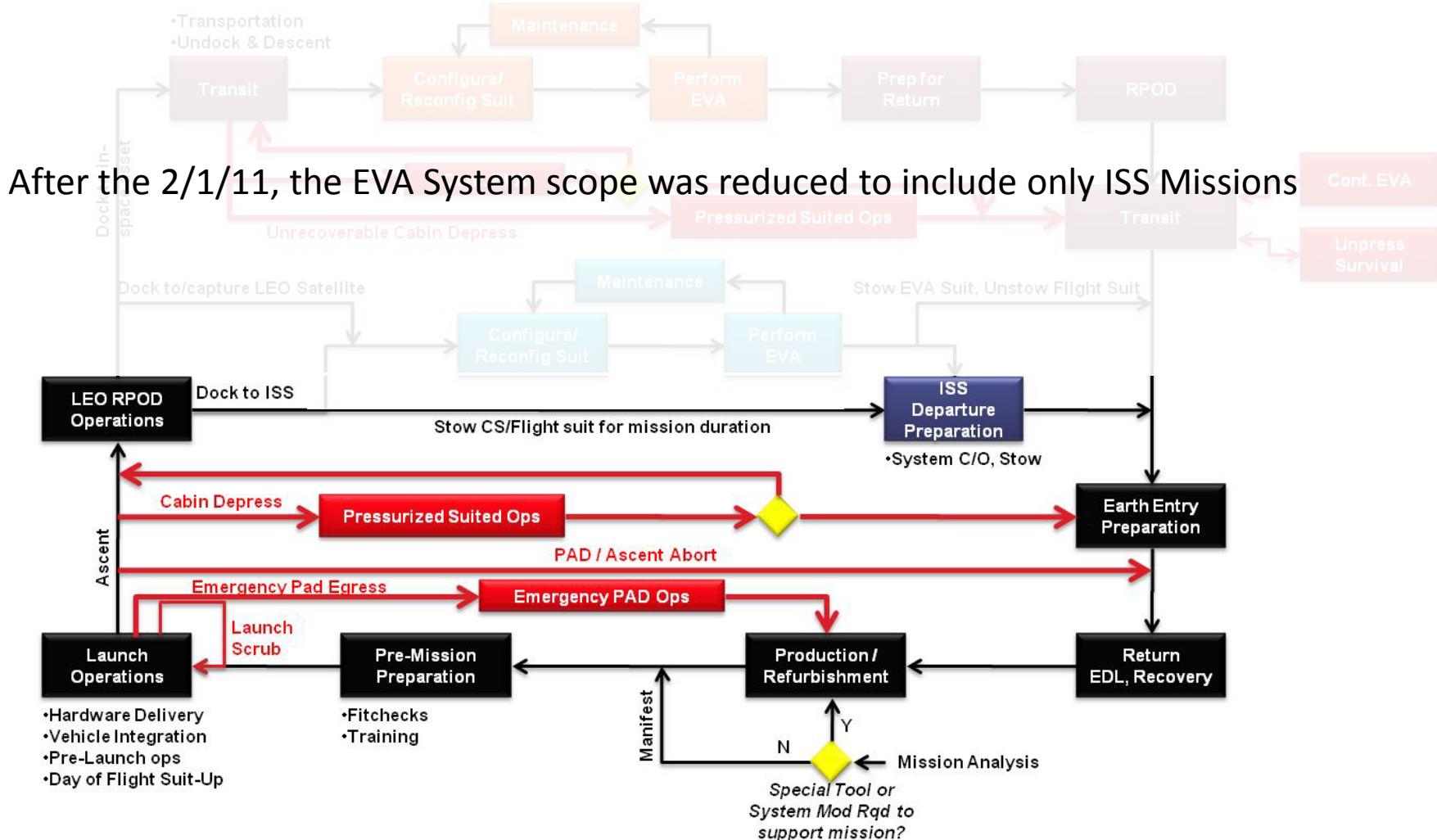
THE SPACE SUIT MBSE APPROACH PUT TO THE TEST

In response to the announcement of the cancellation of Constellation (2/1/10), the EVA Systems Engineering & Integration team was given the challenge to reduce contract scope commensurate with the redirection and minimize overhead associated with engineering the system

Initial Scope of the Mission



Updated Scope



EVA System Re-Architecting



- An SE&I Tiger Team was stood up and given three weeks to update the technical baseline per the reduced scope:
 - Update to the operational concepts
 - Identify the relevant requirements from Level 2 (CARD), Level 3 (SRD), Level 4 (ERDs), HSIR, and IRDs
 - Updating the deliverable components list
 - Update to technical resource allocations (mass, volume)
 - Update verification responsibility
 - Update the Applicable Documents List
- Reports of requirements with mission phase applicability and parent-child relationships identified helped to expedite this process
- Existing models defining mass helped to expedite update to mass allocations

Model Updates



- Changes to the model instigated by this effort:
 - Implemented linkage of requirements to hardware configurations instead of mission phase “technical points of contact”
 - Separating out of functional requirements with mission / configuration dependent performance specs
- Documents can be generated by printing out requirements linked to certain architecture items
- Without changing the current structure, these additional links will allow creation of LEA-focused documents:
 - J-19
 - SRD
 - ERDs

Requirements Streamlining Effort



- In order to minimize systems engineering overhead, a restructuring effort was undertaken to reduce the number of requirements in the EVA System requirements set; this would save resources by reducing unnecessary requirements management and verification activity
 - Delete redundant requirements
 - Merge related requirements where appropriate
 - Goal was to maintain each “requirement” only once in the entire technical baseline; i.e. Do not duplicate an Interface Requirement Document (IRD) requirement in the SRD
- At this time, the requirements were also leveled
 - We took the existing SRD and 2 Element Requirement Documents (ERDs) and combined them into one document. Goals of this effort included the following:
 - Levied the requirements at the level appropriate for verification, i.e. pushed Contract End Item (CEI) specific requirements down to the CEI Specification Documents
- We updated the model framework to produce a requirements set that is easily extensible to future suit developments
 - Established a set of generic space suit standards in the EVA SRD

Requirements Leveling Approach



- EVA SRD System-level requirement
 - Functions that include multiple elements
 - Standards that apply to multiple suit developments
- EVA SRD Element-level requirement
 - Decompositions specific to suit development
 - Allocations where Project-level margins are desired
 - Requirements necessary to bound contract scope
- LEA Suit ERD-level requirement
 - Decompositions that are appropriate for the Element-level
 - Element-level details where contractor ownership is desired
- Subsystem-level requirement
 - Section removed; allocation directly to CEIs from the Element-level
 - Rely on the assignment of technical owners to track responsibility for applicable requirements

Push to Data-Centric



- The EVA Systems Engineering Model inherently captures the SE definition in a data centric manner (as opposed to document-centric)
- Through use of publishing software, this data is sorted and presented in document format per the customers requirements
- A data-centric configuration management approach was developed but never implemented by the EVA Office due to resource constraints
- Things yet to be considered
 - Linkage to applicable documents
 - Implementation of document-centric requirements and other systems definition artifacts applied by the Program or other authoritative sources



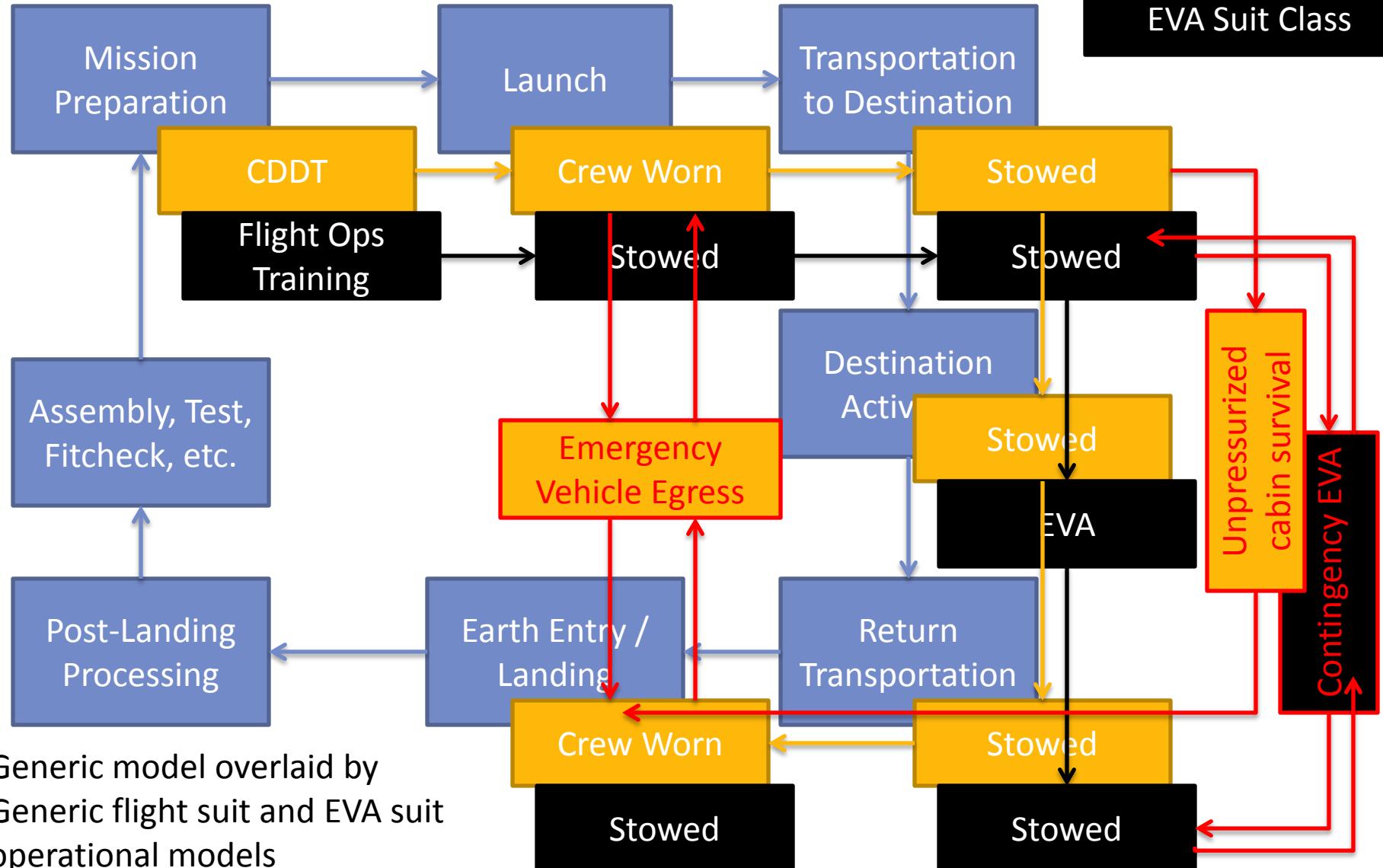
NEXT STEPS

Develop “Generic” Models

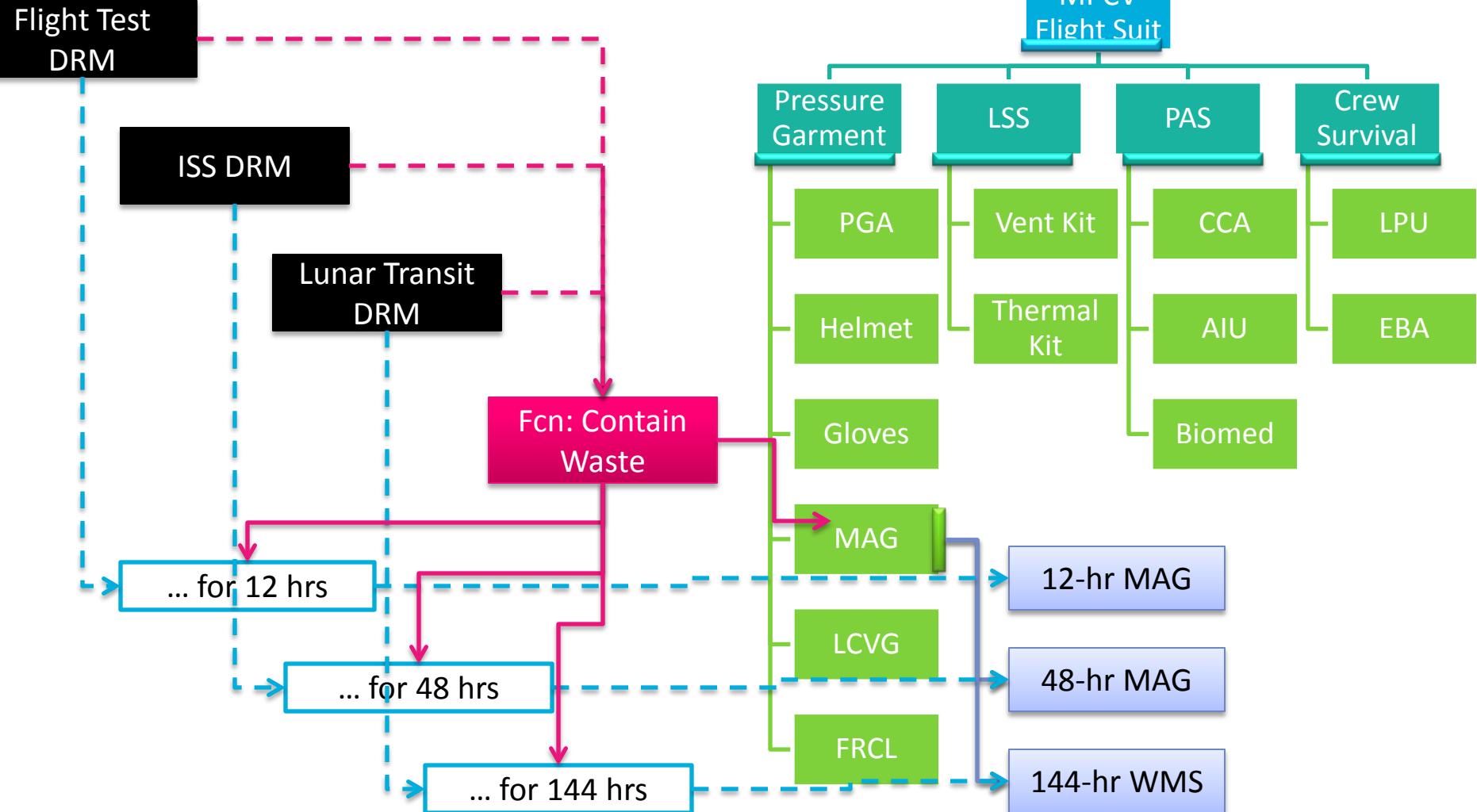


- Utilize the system model developed for Constellation to develop a generic space system model
- Development of a generic model would reduce “start-up” cost and time and time for a new program
 - Basic operational concepts, capabilities and even requirements would already be defined
 - The time to establish the data infrastructure would be significantly reduced
- A generic system model would aid in mission architecting studies
 - Basic capabilities would be defined to support high level functional analyses
 - Basic operational concepts would be defined to support high timeline analyses
 - Resource utilization could be defined to support trade studies and help size supporting systems
- Capturing basic design features will help lead to suit design standards

Generic Operational Models



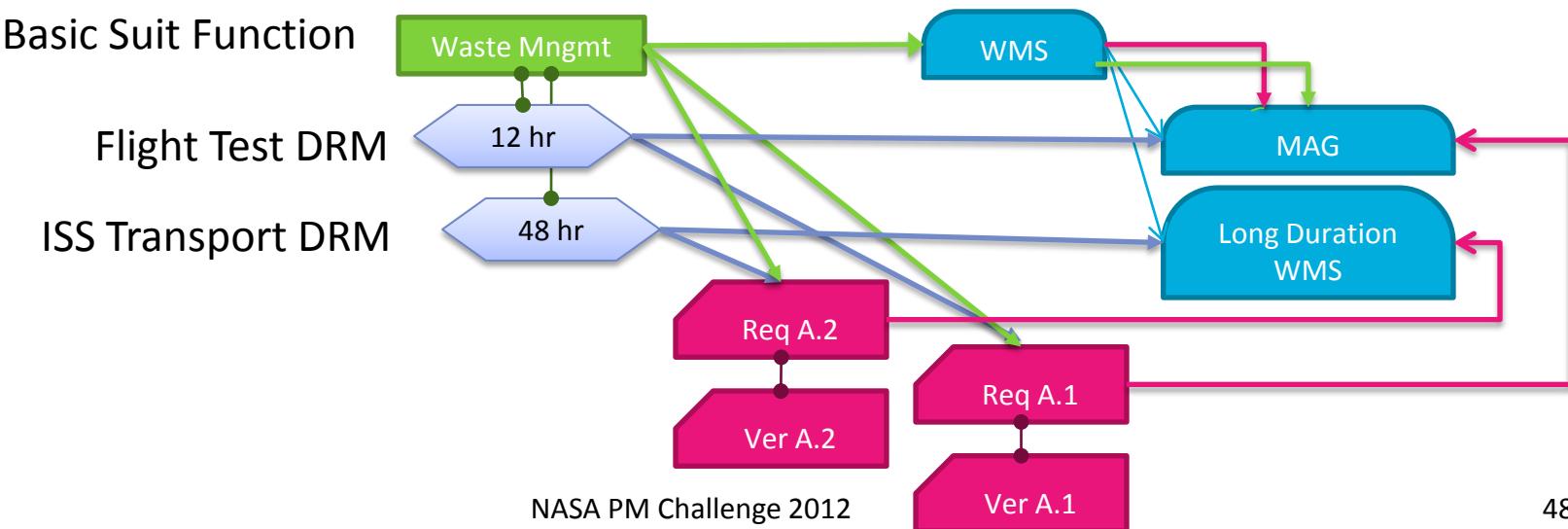
Update Model to Capitalize on Class Definitions



Performance Items to be Implemented



- We realized that multiple requirements were required for each function based on the specific performance required for a particular mission / program
- We are currently exploring the best method for implementing performance items
 - Performance items would link to functions in accordance with specific ops con
 - Functions would continue to link to architecture items, and performance specifications would be linked to specific configurations



Data-centric Configuration Management



- Control data not documents
- Allows contextual data (e.g. concepts of operations, functional analysis, etc) to be reviewed along with controlled data such as requirements
- Data objects can be published in various reports customized to the stakeholder without compromising the integrity of the data
- Allows for more comprehensive stakeholder review
- Shifts emphasis on content of the data instead of the scope of a document



LESSONS

Lessons Experienced...



- Tools and processes for complex systems need to be architected just as carefully as the hardware being developed
 - Want to implement something before folks get comfy
- Fancy tools are only useful if people use them
 - Grandfathered processes
 - Parallel processes
- Data-centric is great... in theory
 - People think in documents
 - Determination of what data needs to be controlled is not as intuitive as with documents
- Communication interfaces are just as important as the data being provided
 - Not everyone needs to handle the data in its native/source environment
 - We need to show it the way that people can read it

... time will tell if these lessons were learned



QUESTIONS



BACK-UP

Mass Tracker Spreadsheet



	A	B	C	D	E	F	G	H
1	Elements:	CSSE	VIE	TEE		Equipment Item	Element	Subsystem
2	PGS	Umbilical	FSE		(TSE) 1-g Internal Harness/Load Distribution	CSSE	Suit_GSE	
3	PCAI	Suit_IF	EVA_TE		GSE_Audio_Checkout	CSSE	Suit_GSE	
4	PLSS							
5	Suit_G							
6								
7								
8								
9								
10								

MEL and MGA multipliers feed the component Mass spreadsheet

	A	B	C	D	E	F	G	H	I	P
Items Details										
1	Element	Subsystem	Equipment Item	QTY	BME, kg	Ibm				
21	CSSE	PLSS	Secondary Oxygen Assembly	0						
22	VIE	Orion_IF	Suit_Suit Interface Assembly	0						
23	VIE	Suit_IF	Suit Multiple Connector Assembly (integrated in suit)							
24	VIE	Suit_IF	Mate_Before_Break Assembly (integrated in suit)							
25	TEE	EVA_TE	Microgravity EVA Tools Harness	1						
26	TEE	EVA_TE	Microgravity EVA Equipment	1						
27	TEE	EVA_TE	EVA Vehicle Maintenance Tools	1						
28	Partially Born Equipment									
29	VIE	Umbilical	IVA Life Support Umbilical Assembly							
30	VIE	Umbilical	EVA Life Support Umbilical Assembly							
31	Suit-Born Consumables									
32			LCG Water	5						
33			IVA LSUS Water							
34			EVA LSUS Water							
35										
36										
37										
38										
39										
40										
41										
42										
43	External System Suit-Born Items									
44	System	Equipment Item	QTY	BME	Ibm	PMEM	BME	Ibm	PMEM	QTY
45	FCE	Beacon	1							
46	FCE	Flare	1							
47	FCE	Crew Options	1							
48	Orion	Vehicle Maintenance Tools	0							
49										
50										
51										
52										
53										
54										
55										
56										
57										
58										

Mass Spreadsheet feeds manually assembled configuration spreadsheets

Configuration totals feed a page which contains historical data

We want to get this in the SE database as soon as we can get it to do math!!!

1	Element	Subsystem	Equipment Item	Hardware Type	Dry Mass	Unit	Mass, kg	Mass, Ibm	Estimate	Rationale	Maturity	MGA	MGA, kg	MGA, Ibm	PME	PME, kg	PM
2	CSSE	PCAI	Biomedical Data Acquisition Assembly	Electronics < 5kg		kg			GRC Bottoms-up estimate	Estimated	30%						
3	CSSE	PCAI	Camera Assembly	Electronics < 5kg		kg			Bottoms-up estimate	Estimated	30%						
4	CSSE	PCAI	Caution, Warning, and Control Assembly	Electronics < 5kg		Ibm			Bottoms-up estimate	Estimated	30%						
5	CSSE	PGS	Communications and Navigation Assembly	Electronics < 5kg		Ibm			Bottoms-up estimate	Estimated	30%						
6	CSSE	PGS	Communications Assembly Communications Carrier Assembly	Softgoods		Ibm			ACES SCUP	GFE/Spec Value	0%						
			Crew Survival Features Emergency Breathing	Softgoods		Ibm			ACES EOS and LPU-2lbs for non-parachute harness	Existing Hardware	3%						



TPM graph is generated from the historical data worksheet

Design Compliance



- Requirements Design Compliance purpose is to objectively determine if the preliminary design can be expected to meet requirements to an acceptable level of risk
- The objectives of requirements design compliance include
 - Identify and establish resolution paths for architectural performance/design issues as early in the design cycle as possible
 - Proactively manage 'acceptable level of risk'
 - Early issue resolution/prevention reduces cost and schedule impacts
 - End-to-End mission perspective
 - Utilize objective design evidence to determine success of design cycle, from architecture perspective of the design reference mission (DRM) and operations concepts (ops con)
 - Facilitate vertical integration of design compliance data with Projects and horizontal integration across level II
 - Report and track significant design compliance issues (design compliance matrix is only one part of design compliance)
 - Engage architecture requirement owners in design aspects in preparation for requirement verification ('get our hands dirty')

Verification Logic Network (VLN)



- VLN is:
 - A strategy to provide evidence of closure for each Detailed Test Objective (DTO)
 - Bottoms-up plan by which verification events can be efficiently executed (ensures efficient grouping of requirements)
 - Will logically group multiple requirements that can be verified with one activity
 - Verification event can be a DAC cycle closure, MEIT, FEIT, DSIL event, Flight Test, Mission Sim
 - Graphical representation of the verification events to close out the applicable requirements set (i.e. CARD, IRD, SRD)
 - Aid in identifying gaps and/or any overlaps in the verification planning process
 - Schedule conflicts
 - Delivery conflicts with hardware or software
 - Clarify test configurations and processes
- In summary, a VLN is an integrated system verification activity network